



---

**VisiCalc<sup>®</sup>**

---

**Owner's Manual**

---

**Series 70**

VisiCalc® designates a software product that is produced under license from VisiCorp and Software Arts, Inc.

VisiCalc® is a registered trademark of VisiCorp.

### **Notice**

**Hewlett-Packard Company, VisiCorp, and Software Arts, Inc. make no express or implied warranty with regard to the program material offered or the merchantability or the fitness of the program material for any particular purpose. The program material is made available solely on an “as is” basis, and the entire risk as to its quality and performance is with the user. Should the program material prove defective, the user (and not Hewlett-Packard Company, VisiCorp, Software Arts, Inc., nor any other party) shall bear the entire cost of all necessary correction and all incidental or consequential damages. Hewlett-Packard Company, VisiCorp, and Software Arts, Inc. shall not be liable for any incidental or consequential damages in connection with or arising out of the furnishing, use, or performance of the program material.**





**VisiCalc®**

# **Owner's Manual**

**For Use With the HP-75**

**May 1983**

00075-90102



## Introducing HP-75 VisiCalc®

HP-75 VisiCalc® provides the analytical power and flexible operation of the VisiCalc® program in a portable personal computer. With the VisiCalc module installed, your HP-75 is transformed into a dynamic problem-solving tool that will handle problems ranging from business and personal financial analysis to engineering and scientific analysis. As you learn to use VisiCalc, you will discover it to be a friendly companion that can assist you in all types of problem-solving and decision-making tasks.

The best way to learn to use VisiCalc is to try it. Don't be afraid to experiment and make mistakes. You can't hurt the computer or the VisiCalc program, no matter what you type on the keyboard. As you practice, you'll gain familiarity and confidence in using the more advanced features, which will speed your work further.

You can learn the elementary features of VisiCalc in an hour or two, and you'll find that you are immediately able to solve simple problems. As you progress through this manual you'll find that VisiCalc has a broad range of features and commands that can make it a powerful problem-solving tool.

In "How To Use This Manual," on page 7, you will find information on how best to approach your study of this powerful new software tool.

# Contents

<b>How To Use This Manual</b> .....	<b>7</b>
For Users With VisiCalc Experience — For Users With No VisiCalc Experience	

## Part I: VisiCalc® Fundamentals

<b>Section 1: Getting Started</b> .....	<b>15</b>
Contents — What You'll Learn in This Section — Installing and Removing the VisiCalc Module — What Is VisiCalc? — A VisiCalc Example — More About Entering the VisiCalc Program — The VisiCalc Display — The Status Line — Accessing Cells — VisiCalc Operational Levels — Top Level — Edit Level — Command Level — Exiting the VisiCalc Program — Returning To Top Level — Memory Management — VisiCalc System Requirements — Where Do We Go From Here?	
<b>Section 2: Creating and Editing Worksheets</b> .....	<b>53</b>
Contents — What You'll Learn in This Section — Headers — Creating and Editing Cell Entries — Repeating Labels — Using Redefining Keys — Modifying the Cell Display — Column or Row Manipulations	
<b>Section 3: Using Functions and Formulas</b> .....	<b>69</b>
Contents — What You'll Learn in This Section — Expressions and Numbers — VisiCalc Functions — HP-75 Functions and VisiCalc — Precalculation of an Expression — Recalculating the Worksheet	
<b>Section 4: Using the Replicate Command</b> .....	<b>85</b>
Contents — What You'll Learn in This Section — The Replicate Command Procedure — Replicating Numbers and Labels — Replicating a Formula — Other Types of Replication	

## Part II: Using VisiCalc® With Peripheral Devices

<b>Section 5: Storing and Retrieving Worksheet Files</b> .....	<b>103</b>
Contents — What You'll Learn in This Section — Using Magnetic Cards — Using a Mass Storage Device	
<b>Section 6: Using a Printer with VisiCalc®</b> .....	<b>105</b>
Contents — What You'll Learn in This Section — Connecting and Assigning the Printer — Using Other Printers — Using the Print Command — Printing the Worksheet Formulas — The Effect of Formats and Column Widths on Printed Worksheets	

<b>Section 7: Using a Video Display with VisiCalc®</b> .....	<b>119</b>
Contents — What You'll Learn in This Section — Connecting and Assigning the Display — Activating and Deactivating the Video Display — Using the Video Display — The Effect of Formats and Column Widths on the Video Display	
<b>Section 8: Formatting the Worksheet</b> .....	<b>131</b>
Contents — What You'll Learn in This Section — Local Formatting — Global Formatting — Local Graphing Format — Overflow Display	

## Part III: Advanced VisiCalc® Operations

<b>Section 9: Accessing Another Worksheet From a Cell</b> .....	<b>143</b>
Contents — What You'll Learn in This Section — External Cell References — Specifying a Single Cell in Another Worksheet — Specifying a Range of Cells in Another Worksheet — Inconsistent Worksheets With External References	
<b>Section 10: Extending the VisiCalc® Function Set</b> .....	<b>151</b>
Contents — What You'll Learn in This Section — What Is an Extension Function? — Accessing an Extension Function From a Cell — Passing Values To and From Extension Functions — Creating Your Own Extension Functions — LEX Files in RAM or ROM	

## Appendixes and Indexes

<b>Appendix A: Owner's Information</b> .....	<b>169</b>
Contents — Maintenance — Limited One-Year Warranty — Service — Technical Assistance — Dealer and Product Information	
<b>Appendix B: VisiCalc® Functions</b> .....	<b>175</b>
<b>Appendix C: The IRR Extension Function</b> .....	<b>178</b>
<b>Appendix D: Changing to Foreign Language Prompts</b> .....	<b>181</b>
Contents — Introduction — Prompt Substitution — The Set of Message Elements — Translation Considerations — An Example VISIMSGS File	
<b>Appendix E: Setting the Video Display Parameters</b> .....	<b>186</b>
Contents — Introduction — Setting the Video Length — Setting the Video Width	
<b>Appendix F: Glossary</b> .....	<b>188</b>
Contents — Special Symbols — Terms	
<b>Command and Operation Summary and Index</b> .....	<b>201</b>
Contents — Start-Up Keys — Top Level Keys — Edit Level Keys — Command Level	
<b>Subject Index</b> .....	<b>211</b>



## How To Use This Manual

This manual is both a learning and a reference tool: It is designed to enable you to rapidly and easily learn to use all of the features of VisiCalc and it is intended to serve as a comprehensive reference to the full set of VisiCalc commands and operations.

The VisiCalc manual describes the operation of the VisiCalc program. Before you read this manual or try to use VisiCalc, you should read and understand part I of the *HP-75 Owner's Manual*. The information provided in the three sections of that part will provide you with the fundamental knowledge that will enable you to get the most out of the information in this manual.

The division of this manual into parts reflects how the information in each section is related to the information in other sections in the part. Part I (“VisiCalc Fundamentals”) presents the basic information you need to use VisiCalc with your HP-75 portable computer. Part II (“Using VisiCalc With Peripheral Devices”) provides information on expanding the use of VisiCalc through the use of HP-75 compatible peripherals. Part III (“Advanced VisiCalc Operations”) covers advanced features of HP-75 VisiCalc. Because of the advanced nature of the material in part III, we recommend that you not attempt to use the features described there until you are thoroughly familiar with VisiCalc. The paragraphs below describe the content and purpose of the individual sections in the manual.

Section 1 (“Getting Started”) introduces you to the fundamental operations of HP-75 VisiCalc. By the time you complete section 1, you should be able to begin using VisiCalc to create, modify, and use your own worksheets.

Sections 2 (“Creating and Editing Worksheets”), 3 (“Using Functions and Formulas”), and 4 (“Using the Replicate Command”) provide more information about the fundamental operations and commands of HP-75 VisiCalc. These sections complete the coverage of the basics of VisiCalc begun in section 1. By the time you complete these three sections you should be able to create, modify, and view your own worksheets, and you should be able to use VisiCalc for a wide range of problems.

Section 5 (“Storing and Retrieving Worksheet Files”) explains how you can use the HP-75 magnetic card reader or an HP-IL mass storage device to provide permanent storage of your worksheet files on magnetic cards or other mass storage media.

Sections 6 (“Using a Printer with VisiCalc”), 7 (“Using a Video Display with VisiCalc”), and 8 (“Formatting the Worksheet”) describe how to use a peripheral printer or video display device with VisiCalc to print or display your worksheets.

Sections 9 (“Accessing Another Worksheet From a Cell”) and 10 (“Extending the VisiCalc Function Set”) introduce and explain two unique and powerful HP-75 VisiCalc capabilities: the *external cell reference* and the *extension function*. Together these two features greatly expand the power and flexibility of VisiCalc.

The appendixes at the back of the manual provide additional reference information about care and service of the VisiCalc module, the VisiCalc function set, how to change the VisiCalc command prompts from English to another language, and how to set the video display parameters. A comprehensive glossary of terms used in this manual is also included.

Two indexes are provided in this manual. A command and operation summary and index provides a detailed summary of all VisiCalc operations. A subject index is also provided to assist you in finding information in the manual when you are using it for reference purposes.

The *VisiCalc Quick Reference Guide* packaged with VisiCalc lists, in condensed form, the operations, commands, and functions of VisiCalc. It is intended for use as a handy reference guide after you have learned to use VisiCalc by reading this manual.

A second manual, the *VisiCalc Programmer’s Reference Manual*, is included with VisiCalc. The programmer’s reference manual describes how you can use the spreadsheet language contained in the VisiCalc module (the VisiCalc program was created using this spreadsheet language) to create customized spreadsheet-like applications. None of the information in that manual is required to use VisiCalc. Before reading the programmer’s reference manual, or attempting to use the information in it, you should thoroughly understand HP-75 VisiCalc and be well-versed in HP-75 BASIC language programming.

An overlay has been included in the package to help you use VisiCalc. The mnemonics on the overlay will remind you of the keyboard location of special operations.

A card pac, containing five HP-75 magnetic cards, is also packaged with VisiCalc. One card contains an example worksheet that you will use in section 1 to explore some of the capabilities of VisiCalc. Another card contains a function that will be used in section 10 to illustrate one of the unique features of HP-75 VisiCalc—the ability to expand the VisiCalc function set by creating your own functions. The remaining cards are described in the *VisiCalc Programmer’s Reference Manual*.

## For Users With VisiCalc® Experience

If you have experience with other VisiCalc programs (like Hewlett-Packard’s Series 80 VisiCalc® PLUS), you will find much that is familiar about HP-75 VisiCalc. Most of the commands, operations, and functions of HP-75 VisiCalc work like their counterparts in other VisiCalc programs. However,



because HP-75 VisiCalc is designed to be used on a portable computer with a single-line display, there are several differences in the way certain operations work in this product. In addition, some operations available on other VisiCalc programs are not implemented on HP-75 VisiCalc. Conversely, HP-75 VisiCalc contains several features that are not available on other VisiCalc programs.

The most significant difference between HP-75 VisiCalc and other VisiCalc programs is that HP-75 VisiCalc was designed for use on a portable computer. It provides you with the full power and versatility of a sophisticated spreadsheet program, combined with the portability and flexibility of your HP-75 computer.

Next, with HP-75 VisiCalc, you normally see only one cell of the worksheet at a time, since it was designed to work with the HP-75's single line display. To make it easier for you to use the worksheet under this condition, a number of features have been designed into the HP-75 VisiCalc program. One of these features, the user-defined header, enables you to replace the usual column (A, B, C, ...) and row (1, 2, 3, ...) headers with alphanumeric labels that identify the contents of a column or row such as, for example, the MARCH column or the EXPENSES row. You can then use these user-defined headers for specifying cell coordinates in commands and in formulas. User-defined headers are very helpful since it might be difficult to remember that MARCH-EXPENSES is in cell C8 or that JULY-INTEREST is in cell G5 when you can only see one cell at a time. With user-defined headers you simply specify MARCH-EXPENSES or JULY-INTEREST as the desired cell coordinates. VisiCalc knows where to find those cells—even if, as a result of changes you have made to the worksheet, MARCH-EXPENSES is now located at cell D8 or JULY-INTEREST has moved to cell H6.

Also, because the continuous memory capability of the HP-75 allows multiple files in memory, VisiCalc has the ability to access the contents of one worksheet from another, and also provides you with the means to extend the built-in function set of VisiCalc to include *extension functions* that you create. Since worksheets are stored as files in memory—and many such files can be in memory at the same time—VisiCalc allows you to place a reference to a value or even to a range of values from other worksheets in a formula. Likewise, a call to a specially formatted BASIC program (an extension function) can be contained in a formula. The ability of one worksheet to access another means that you can create worksheets with a “third” dimension and vastly increased analytical power. The ability to create your own functions and call them from a cell means that you can create special purpose function sets designed for your own specific applications.

Summarized below are the principal differences between HP-75 VisiCalc and other VisiCalc programs:

- The maximum worksheet size is 255 columns (A through IU) by 255 rows (1 through 255), although the practical size of worksheets will be limited by available memory.
- Only one cell at a time is visible in the HP-75 display. (If a video display is connected to the system, the usual multi-column by multi-row matrix is displayed.)
- User-defined alphanumeric column and row headers can be created and used for referencing cells.

- There is no Window command in VisiCalc. HP-75 VisiCalc does have an “alternate viewing window,” which provides a capability similar to the Window command. (The **/W** command in HP-75 VisiCalc is the Local Column Width command.)
- In addition to the ability to set the global column width, local column widths can be separately set for any column in the worksheet.
- Worksheets are stored in memory as files, and many such worksheet files may reside in memory simultaneously. A formula in one worksheet can access a cell or a range of cells in another worksheet in memory.
- In addition to worksheet files, other HP-75 files of all types and in any number (limited by memory capacity) can also be in memory.
- You can create *extension functions* that expand the set of built-in VisiCalc functions. These functions reside in the HP-75’s user memory and you access them in the same way that you do built-in functions.
- The HP-75 VisiCalc **/P** (*Print*) command can print the worksheet either with or without the column and row headers. The Print command can also print a list of the formulas used in the worksheet.
- HP-75 VisiCalc does not contain a Store command, but worksheet files can be stored to and retrieved from both magnetic cards and mass storage using the HP-75 **COPY** command.
- HP-75 VisiCalc does not provide the ability to “point” the cursor so as to place a cell reference in a cell.
- HP-75 VisiCalc does not allow the use of the arrow keys to terminate cell entry.
- HP-75 VisiCalc does not provide a Blank command. Cells are cleared by pressing **CLR** **RTN**.
- HP-75 VisiCalc does not provide a Clear command. Worksheets are cleared by using the HP-75 **PURGE** command to remove the worksheet file from memory.
- HP-75 VisiCalc does not have a Title command. User-defined headers provide much of the utility of a Title command, and they provide the additional ability to reference cells using them as described above.
- HP-75 VisiCalc does not have Version command. The **/V** command in HP-75 VisiCalc is the Video command, which activates and deactivates the video display.
- HP-75 VisiCalc does not have an Edit command. Previously entered cell contents are edited by pressing the HP-75 **EDIT** key.
- The HP-75 VisiCalc program is entirely contained in a plug-in module. Because of this, most of the HP-75’s user memory (RAM) is available for the storage of worksheet files and other user files. (The HP-75 operating system and VisiCalc combined use about 4500 bytes of memory for system overhead.)

The list below indicates those parts of this manual that you should read to learn about a particular operation or class of operations.

The list below indicates those parts of this manual that you should read to learn about a particular operation or class of operations.

To Learn About:	Read:
Creating, editing, and viewing worksheets.	Sections 1 and 2.
Using VisiCalc or HP-75 functions with worksheets.	Section 3 and appendix B.
Using the Replicate command.	Section 4.
Storing worksheet files on mass storage media.	Section 5.
Printing worksheets.	Sections 6 and 8.
Using a video display with VisiCalc	Sections 7 and 8 and appendix E.
Accessing cells in another worksheet.*	Section 9.
Using and creating extension functions.*	Section 10.
* These two concepts are advanced features unique to HP-75 VisiCalc. You should be familiar with the information presented in part I, "VisiCalc Fundamentals," before trying to perform these operations.	

## For Users With No VisiCalc® Experience

If this is your first contact with an “electronic spreadsheet” program, you will find that VisiCalc is easy to learn and even easier to use. However, as with all powerful and flexible tools, you will not be able to fully utilize VisiCalc’s power and flexibility unless you invest some time in learning to use it. We recommend that, before trying to use VisiCalc to any extent, you take the time to carefully read the content of—and work through the examples in—part I, “VisiCalc Fundamentals,” and that you also read through section 5. You should read the applicable sections in part II, “Using VisiCalc With Peripheral Devices,” if you intend to use a printer or video display device with VisiCalc. Part III, “Advanced VisiCalc Operations,” presents advanced concepts unique to HP-75 VisiCalc. We suggest that you read through the sections in this part to get an idea of their content, but you should master the fundamental concepts presented in part I before attempting to use these advanced capabilities.



# **Part I**

## **VisiCalc<sup>®</sup> Fundamentals**

**Part I** contains the basic information you need to use HP-75 VisiCalc. If you have no experience using a spreadsheet program like VisiCalc, you should read part I and work through the examples in each section to learn how to use the basic features of VisiCalc. If you have used a VisiCalc program before, part I will show you how the operation of HP-75 VisiCalc differs from other VisiCalc programs.

**Part I** contains the following sections:

- **Section 1** (“Getting Started”) introduces the fundamental operations of VisiCalc and gives an overview of its basic capabilities. In this section you will learn how to install and remove the VisiCalc module, enter the VisiCalc program, view the status line, access cells in the worksheet, make cell entries, and exit VisiCalc. You will also learn about the cell display, VisiCalc’s operational levels, memory management, and system requirements.
- **Section 2** (“Creating and Editing Worksheets”) provides more information about creating, modifying, and viewing worksheets. In this section you will learn how to create user-defined headers, edit cell contents, create repeating labels, use redefined keys, modify the cell display, and insert, delete, or move columns and rows.
- **Section 3** (“Using Functions and Formulas”) describes the available VisiCalc functions and how to use formulas effectively. In this section you will learn how to create arithmetic expressions, use VisiCalc functions, precalculate an expression, and manually recalculate the worksheet.
- **Section 4** (“Using the Replicate Command”) describes one of the most powerful and useful VisiCalc commands—the Replicate command. In this section you will learn how to replicate a single cell or a range of cells containing numbers, labels, or formulas.

## Section 1

# Getting Started

## Contents

What You'll Learn in This Section .....	16
Installing and Removing the VisiCalc Module .....	16
What is VisiCalc? .....	17
A VisiCalc Example .....	19
More About Entering the VisiCalc Program .....	24
The VisiCalc Display .....	27
The Cell Display .....	28
Headers .....	30
The Status Line .....	30
Accessing Cells .....	32
Using the Arrow Keys .....	32
Using the Go To Command .....	33
The Alternate Viewing Window .....	34
VisiCalc Operational Levels .....	36
Top Level .....	37
Edit Level .....	38
Labels .....	39
Values .....	41
Formulas .....	42
Functions .....	45
Extension Functions .....	45
Accessing Other Worksheets in Memory .....	46
Command Level .....	46
Exiting the VisiCalc Program .....	47
Returning to Top Level .....	47
Memory Management .....	47
Dynamic Worksheet Size .....	47
Multiple Files in Memory .....	49
Purging Worksheet Files .....	49
VisiCalc System Requirements .....	50
Where Do We Go From Here? .....	51

## What You'll Learn in This Section

In this section you will learn the basic operations needed to begin using VisiCalc. When you finish this section, you should be able to:

- Install and remove the VisiCalc module.
- Enter the VisiCalc program and create or access worksheets in memory.
- Examine the cell display and the worksheet by using the various display operations and cell display movement operations.
- Understand the distinction between Top level, Edit level, and Command level, know how to move between levels, and know what types of operations can be performed at each level.
- Recognize the different types of cell entries, and enter labels, values, formulas, external cell references, functions, and extension functions.
- Recognize the 11 different VisiCalc commands and know, in general, what operation each performs.
- Exit the VisiCalc program.
- Understand how VisiCalc manages worksheet files in HP-75 memory.

As you work through this section, remember that many of the operations introduced here will be discussed in more detail later in the manual. The purpose of this section is to introduce you to the fundamental operations of VisiCalc and to provide you with the essential knowledge necessary to begin using VisiCalc.

## Installing and Removing the VisiCalc Module

The VisiCalc module can be plugged into any of the three ROM ports on the front edge of the computer.

### CAUTIONS

- Be sure to turn off the HP-75 (press **SHIFT** **ATTN**) before installing or removing a module. If there are any pending appointments, type `alarm off` **RTN** in EDIT mode to prevent the arrival of future appointments (which would cause the computer to turn on). If the computer is on or if it turns itself on while a module is being installed or removed, it might reset itself, causing all stored information to be lost.
- Do not place fingers, tools, or other foreign objects into any of the ports. Such actions could result in minor electrical shock hazard and interference with pacemaker devices worn by some persons. Damage to port contacts and internal circuitry could also result.



To insert the VisiCalc module, orient the module so that the label is right side up, hold the computer with the keyboard facing up, and push in the module until it snaps into place. During this operation be sure to observe the precautions above.



To remove the module, use your fingernails to grasp the lip on the bottom of the front edge of the module and pull the module straight out of the port. Install a blank module in the port to protect the contacts inside.

Once you install the VisiCalc module, you will probably want to leave it installed most of the time. The module does not use any battery power, and it has no effect on the normal operation of your computer. The VisiCalc module uses about 90 bytes of memory to store certain status information, so the memory available for program, text, and worksheet files is reduced by that amount whenever the module is installed. When the module is unplugged, this memory is reclaimed by the HP-75 and the status information is lost. (When VisiCalc is actually running, it uses about 2500 bytes of memory for system overhead.)

## What Is VisiCalc®?

VisiCalc was born of the observation that many problems are commonly solved with a calculator, a pencil, and a sheet of paper—three nearly universal tools. Calculating sales projections, income taxes, financial ratios, your personal budget, engineering changes, cost estimates—even balancing your check-book is done with these tools.

VisiCalc gives you a way to use your HP-75 portable computer to solve complex problems without having to spend time and energy writing programs. You can use VisiCalc much as you would a calculator, a pencil, and a sheet of paper—but thanks to VisiCalc’s “electronic worksheet” and recalculation capability, you can work more rapidly and explore options more quickly than you could before.

VisiCalc combines the convenience and familiarity of a pocket calculator with the powerful memory and display capabilities of your portable computer. With VisiCalc, the computer’s display becomes a “window” that looks into the much larger electronic worksheet. You can move this window around the worksheet in a variety of ways to examine or change any part of the worksheet.

The VisiCalc worksheet is organized as a grid of columns and rows. The intersections of the columns and rows define thousands of entry positions (called cells). You can enter into any cell a number, a formula to be calculated, or an alphabetic label. Simply by “writing” on the sheet, you can set up your own charts, tables, and records. Formatting commands let you customize the appearance of each entry, column, or row. For example, if you wish you can create a VisiCalc spreadsheet that looks just like the statement you get from your bank.

But the power of VisiCalc is not that it simply imitates a calculator, a pencil, and paper. The real power of VisiCalc comes from the fact that when you do a calculation to determine the value to put in a cell, VisiCalc can remember those calculations and store them—along with the calculated result—in the cell. These calculations are stored in the cell as *formulas*, and formulas can contain references to other cells in the worksheet. So if you change a value you had previously written on the electronic worksheet, all the other numbers that are related to that value by a formula will be changed too, as VisiCalc automatically recalculates all the formulas on the worksheet.

Recalculation makes VisiCalc a powerful planning and forecasting tool. Not only can you effortlessly correct mistakes and omissions, you can also examine various alternatives. For example, imagine that you are doing sales projections using VisiCalc. You might want to know what the impact on your company will be if a specific product doesn’t sell as well as anticipated. What if you sell only 200 widgets a month instead of 250? What if you sell 300? What if one of your salespeople quits and it takes a replacement six weeks to come up to speed? Playing “what if” with VisiCalc is simply a matter of changing a single number, while doing the same thing with a calculator, pencil, and paper might take hours of erasing and recalculating.

VisiCalc’s editing features let you change, insert, delete, move, or copy numbers, formulas, or labels. For example, you can add, remove, or move whole columns or rows; you can copy the information in a cell or a group of cells into other cells; and you can control the display format of the information contained in cells. As you make these changes, VisiCalc automatically restructures the worksheet to reflect your modifications.

The ease of entering and editing information, coupled with the powerful formatting commands, make it easy to create customized “applications software” without doing any programming. For example, a financial advisor might prepare a financial analysis worksheet that could then be used as a “template” into which a client’s specific financial situation could be entered for analysis. The range of problems that can be attacked with VisiCalc is not limited to the business or financial world—it includes a wide range of engineering and scientific applications as well. For example, an engineer could create a template that could be used to analyze the energy budgets of different buildings.

The paragraphs above give a mere hint of the enormous potential for problem-solving contained in VisiCalc. In the rest of this manual you will explore that potential and learn how to use it for your own applications.

## A VisiCalc® Example

VisiCalc can be thought of as a new and powerful HP-75 operating mode analogous in many ways to the HP-75 TIME, APPT, and EDIT modes. Once you enter VisiCalc “mode,” the functions of many of the HP-75’s system and editing keys change, just as they do when, for example, you change from EDIT to APPT mode. Keep this idea in mind as you work through this manual; it will help you to understand how key functions differ between VisiCalc and the standard HP-75 modes.

If you read “What Is VisiCalc?” (page 17), you know a little about the power and flexibility of VisiCalc. Like most people, however, you are probably anxious to get a “feel” for how VisiCalc works and what it can do for you. So, to begin our exploration of VisiCalc, let’s look at a brief example that will illustrate some of its features and operations.

**Note:** The purpose of this example is simply to illustrate some of the features of VisiCalc. The new terms, concepts, and operations introduced here will be fully explained later in this section or in subsequent sections of the manual. To get the most out of this example, just perform the operations as described and concentrate on observing the results of those operations.

One VisiCalc application that should be familiar to everyone is a personal budget that shows expected income and expenses by month. A budget can be set up as a grid of columns and rows, with the names of the months across the top and categories of income and expense down the side. A simplified budget might look, in part, like this:

A Personal Budget

	January	February	March	April	May	June
<b>Income</b>	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00
<b>Housing</b>	400.00	400.00	400.00	400.00	400.00	400.00
<b>Insurance</b>	100.00	100.00	100.00	100.00	100.00	100.00
<b>Taxes</b>	600.00	600.00	600.00	600.00	600.00	600.00
<b>Food</b>	200.00	175.00	180.00	155.00	195.00	140.00
<b>Utilities</b>	110.00	95.00	87.00	81.00	68.00	55.00
<b>Total Expenses</b>	1410.00	1370.00	1367.00	1336.00	1363.00	1295.00
<b>Remaining Income</b>	590.00	630.00	633.00	664.00	637.00	705.00

A worksheet that contains the same information as above is recorded on a magnetic card that is packaged with VisiCalc. Find the card labeled BUDGET in the VisiCalc card pac and load it into memory by executing a copy-from-card command. If you are unfamiliar with card reader operations, please refer to section 8, “Card Reader Operations,” in the *HP-75 Owner's Manual* for a complete description of the procedures for loading the contents of a magnetic card into memory.

After you load the BUDGET worksheet into memory, you can follow the steps described below to examine the worksheet. If your results differ from those shown, carefully review the steps and try them again (you might have missed a step or pressed different keys than those specified).

To enter the VisiCalc program, make sure you are in HP-75 EDIT mode (VisiCalc is only accessible from EDIT mode—either the BASIC or text prompts may be present), type `vc` or `VC`, then press `[RTN]`. The following message will be displayed for about one-half second:

VisiCalc (R) - copyright HP 1983	The VisiCalc start-up message.
----------------------------------	--------------------------------

Then VisiCalc will display the *worksheet name template*.

Enter worksheet name: BUDGET	The worksheet name template.
------------------------------	------------------------------

Now press the `[RTN]` key to make the BUDGET worksheet the *active* worksheet. VisiCalc will respond by displaying the message *Working* briefly (the *Working* message may stay in the display for several seconds for large or complex worksheets),\* then one of the worksheet's cells will be displayed:

Working Jan^Income: 2000.00
--------------------------------

If you look back to the budget shown at the beginning of this section (page 19), you will see that Jan^Income corresponds to the January and Income labels on the budget listing, and that 2000.00 corresponds to the amount shown at the intersection of the January column and the Income row. This display, showing the location and contents of the cell, is called the *cell display*.

**Note:** While you are reading this manual and working the examples, your HP-75 may time-out and turn itself off if VisiCalc remains idle for more than 5 minutes. (VisiCalc keeps track of the `STANDBY` setting of the HP-75. Refer to section 1 of the *HP-75 Owner's Manual* for more information about the `STANDBY` command.) If this happens, you can easily get back to the exact point in the worksheet that you were working on before the time-out occurred. Simply turn the HP-75 back on (press `[ATTN]`), type `vc` `[RTN]`, wait for the worksheet name template, then press `[RTN]` again.

---

\* The *Working* message is displayed during certain VisiCalc operations that causes a perceptible delay in response.

Now press the down arrow key ( $\downarrow$ ); a new cell display will appear:

[Jan]2:

This cell is empty—it contains no information.

Press the  $\downarrow$  key again. Note that each time you press the arrow key the cell display moves to another cell:

Jan^House: 400.00

Look at the colon that separates the label part (called the *cell coordinates field*), and the information part (called the *cell contents field*) of the cell display. In this display, it is underlined. The underlined colon indicates that the value you see in this cell is the *result* of a *formula* that is also stored in that cell. To see what that formula is, press the  $\boxed{\text{APPT}}$  key:

Jan^House: .2\*[Jan^Income]

The  $\boxed{\text{APPT}}$  key is a “toggle” that switches the cell display between *Result Display mode* and *Formula Display mode*.

This formula says that Jan^House is equal to two-tenths of Jan^Income. Because they allow the contents of one cell to affect the contents of another, formulas like this one are fundamental to the flexibility and power of VisiCalc. For example, if Jan^Income changes from 2000.00 to 3000.00, the amount in Jan^House will also change—to 600.00.

Press the right arrow key ( $\rightarrow$ ) and note that Feb^House also contains a formula, and that this formula is related to *February* income:

Feb^House: .2\*[Feb^Income]

If you press the  $\rightarrow$  key a couple more times, you will find that the formula for each month’s housing is relative to the value in the income cell for that month. Press other arrow keys to scroll the cell display around the worksheet. As you do so you will find many formulas relating the contents of a cell to another cell or cells. In fact, by changing just one value—that of January’s income—you can radically alter the appearance of the worksheet.

The worksheet that you loaded from the magnetic card contains the same values as those shown in the budget illustrated at the beginning of the section (page 19). As we change the values in the worksheet in the steps below, refer to that illustration to see how the values in the cells change as a result of your manipulations.



**Note:** The format of the procedure below will be used often throughout this manual to detail a series of keystrokes and resulting displays. In this format, *system* or *editing* keys (example: `[APPT]` and `[↓]`), and *command* keys or keystroke combinations (example: `[/ [M]`) that you must press are shown as actual keycaps. Characters that you should enter into the display are shown as display characters. The displays that are the result of your commands and entries are also shown as display characters, but inside a “display box.”

Remember that the purpose of this section is to get a general overview of VisiCalc. Some of the operations you will perform below will be unfamiliar to you, but don’t worry too much about how the operations work—the important thing at this point is to see the result of those operations.

**Input/Result**

`[APPT]`

`[SHIFT] [>] [Jan^Income] [RTN]`

Jan^Income: 2000.00

3000 `[RTN]`

Working  
Jan^Income: 3000.00

Now use the arrow keys to move around the worksheet and look at the effect of this single change on different cells:

**Input/Result**

`[↓] [↓]`

Jan^House\_ 600.00

`[↓]`

Jan^Insur: 100.00

Sets VisiCalc back to *Result Display mode*.

Executes a command (called the *Go To* command) that moves the cell display back to Jan^Income.

VisiCalc displays the *Working* message because when Jan^Income changes, it takes a few seconds to recalculate all of the cells that are related to Jan^Income.

This value changed because it depends on Jan^Income.

This one didn’t change because there was no formula associated with it. Look at the field separator (the colon); it’s not underlined. (Remember that if the colon is underlined, the displayed value has a formula associated with it.) Press `[APPT]` to verify that there isn’t a formula—the display doesn’t change. Press `[APPT]` again to return to Result Display mode.



```
Jan^Taxes: 900.00
```



```
Jan^Food: 200.00
```



```
Jan^TotExp: 1910.00
```

APPT

```
Jan^TotExp: SUM([Jan^House],[Jan^U
```

TIME

```
A9: SUM([Jan^House],[Jan^Util]
```

TIME

```
Jan^TotExp: SUM([Jan^House],[Jan^U
```

RUN RUN RUN RUN

```
Jan^TotExp: [Jan^House],[Jan^U
```

APPT

```
Jan^TotExp: 1910.00
```

Another value that changes because it has a formula associated with it.

Here's another value that hasn't changed, again because a value was entered, not a formula.

This value also depends on a formula.

Pressing **APPT** sets the display to Formula Display mode. Here's the formula that gave the result above; it contains a new element: a VisiCalc *function*, SUM. Functions provide a way to do more complex computations than can be accomplished with arithmetic operators like + and -. VisiCalc functions will be introduced later in this section and covered in more detail in section 3.

The **TIME** key replaces the *user-defined* headers in the cell coordinates field with *default* headers. When this happens, the cell contents field is increased in size and you can see more of a long formula like the one in this cell.

The **TIME** key is a *toggle*. Pressing it again switches the cell coordinates field back to user-defined headers.

The **RUN** key moves the cell contents one character to the left each time it's pressed. (**SHIFT** **RUN** moves the contents the other way.)

Pressing the **APPT** key sets the display to Result Display mode.

Press the arrow keys a few more times to see how other cells changed as a result of the single change you made to Jan^Income. Although BUDGET is a relatively small and simple worksheet, it still provides a lot of computational power. Just consider the amount of writing and calculating involved in modifying a hand-written worksheet to reflect a change in monthly income. VisiCalc accomplished the same thing with a few keystrokes.

To complete this example and exit VisiCalc, press the **[SHIFT][ATTN]** keystroke combination. When VisiCalc is running, **[SHIFT][ATTN]** exits the VisiCalc program, *but does not turn off the HP-75*.

**Input/Result**

**[SHIFT][ATTN]**

```
Leaving worksheet: BUDGET
>■
```

VisiCalc displays the exit message; the prompt and cursor then reappear. (The text prompt, `,,`, may appear if you entered VisiCalc from HP-75 text mode.)

This brief introduction doesn't begin to cover all of the operations and capabilities of VisiCalc. The new terminology and concepts introduced here will be further explained and defined as you advance through this manual, and additional commands and operations will be added to the few that you used in this section. The remainder of section 1 is devoted to giving you a more thorough introduction to the essential operations of VisiCalc.

**More About Entering the VisiCalc® Program**

As you've already seen, you enter VisiCalc by typing `vc` from HP-75 EDIT mode. In "A VisiCalc Example" (page 19), you then pressed **[RTN]** to call up the BUDGET worksheet. There are several start-up possibilities, however, and now we'll explore them all. Type `vc` to enter VisiCalc again:

```
VisiCalc (R) - copyright HP 1983
```

The VisiCalc start-up message is displayed whenever VisiCalc is activated.

Then VisiCalc will display the *worksheet name template* to prompt for a worksheet file name:

```
Enter worksheet name: BUDGET
```

VisiCalc displays the most recently accessed worksheet if any worksheets exist in memory.

If one or more worksheet files are already in memory, the name field will contain the file name of the most recently accessed VisiCalc worksheet. For example, if you loaded the BUDGET worksheet as described in the previous section, then the worksheet name template will appear as above. If no VisiCalc worksheet files exist in memory, the name field (which begins at the cursor) will be blank and the template will look like this:

```
Enter worksheet name: ■
```

This template is displayed when no worksheets exist in memory.



You can proceed in one of four ways at this point:

- If a worksheet file name is displayed, you can recall it and make it the active worksheet file by simply pressing **[RTN]**.
- You can enter a new worksheet file name by typing the file name in the blank space or over the existing name. When you press **[RTN]**, a new empty worksheet will be created.
- If several worksheet files exist in memory, you can use the arrow keys to scroll through a catalog of worksheet file names.
  - You can recall any worksheet in memory by stepping through the catalog of VisiCalc worksheet files with the **[↓]** key until the name of the desired file is displayed, then pressing **[RTN]**. The files are displayed in reverse of the order of their creation—most recently created first.
  - You can use the **[↑]** key to scroll up through the worksheet catalog after pressing the **[↓]** key one or more times.
  - You can also use the **[SHIFT][↑]** and **[SHIFT][↓]** keys to go directly to the first and last entries in the worksheet catalog, respectively.
- You can exit the VisiCalc program and return to HP-75 EDIT mode by pressing the **[ATTN]** key or **[SHIFT][ATTN]** keystroke. Note that the only time that you can use the **[ATTN]** key to *exit* the VisiCalc program is when the worksheet name template is in the display.

The examples below will clarify the options listed above.

**Example 1:** If a worksheet file named **BUDGET** was already in memory, that file would be displayed in the template. (This is the situation that should exist now if you loaded the **BUDGET** worksheet as described under “A VisiCalc Example.”) You would recall this existing worksheet to the VisiCalc program by pressing the **[RTN]** key.

### Input/Result

Enter worksheet name: BUDGET

[RTN]

Jan^TotExp\_ 1910.00

VisiCalc will display the coordinates and contents of the cell where the cursor was last located in the worksheet. (You may see a different cell—if so press **[SHIFT][>]**, type **[Jan^totexp]**, then press **[RTN]** to see the cell shown.)

**Example 2:** Create a new VisiCalc worksheet by typing a new file name in the name field.

### Input/Result

Enter worksheet name: BUDGET

The worksheet name template as it appears when you first enter VisiCalc.

CLR sales

Pressing CLR clears the name field, so you can enter the new worksheet file name.

Enter worksheet name: sales

RTN

Pressing RTN activates the new worksheet, and the coordinates of the first cell are displayed, ready for cell entry.

A1:

There are several other things that you need to consider with regard to entering new worksheet file names:

- You can use any valid HP-75 file name to name your worksheets—including volatile file names (those which begin with a period).
- If you try to give a worksheet the same name as a non-worksheet file that already exists in memory, the computer will beep, and the worksheet name template will remain in the display to prompt you for another choice of file names.
- If the name you enter is the same as another *worksheet* file name, the named worksheet will be called into the VisiCalc system and no new worksheet will be created. Refer to “Purging Worksheet Files”, page 49, for information about deleting unwanted worksheet files from memory.
- VisiCalc worksheet files are a special file type and are different from other types of files in memory. Worksheet files can be CATALOGUED, RENAMED, PURGED, and COPIED, but not LISTED, PLISTED, EDITED, RUN, CALLED, TRANSFORMED, or RENUMBERED.

**Example 3:** If several worksheet files were in memory, you could use the arrow keys to scroll through these worksheets to find the one you want to work with. For example, suppose the worksheet files BUDGET, SALES, and PAYROLL were in memory, they had been created in the order listed, and the last worksheet accessed was SALES. Then, to find and call up the BUDGET worksheet, you could go through the following process.

After you enter the VisiCalc program, you would see the worksheet file name template:

Enter worksheet name: SALES

The file name that would be shown when the template first appears would be SALES—because it was the last one accessed.

You could press **↓** to see the name of the most recently created worksheet:

Enter worksheet name: **PAYROLL**

PAYROLL was the most recently *created*, but not the most recently *accessed* worksheet. So it would appear after SALES in the catalog.

If you press **↓** again to see the last worksheet file in memory, you would see:

Enter worksheet name: **BUDGET**

BUDGET was the first worksheet created, but the last one in the catalog.

If you press **RTN** now, the BUDGET worksheet would be called into the VisiCalc program. Remember that this example is hypothetical—only the BUDGET worksheet should exist in memory at this time.

**Example 4:** If you enter the VisiCalc program but then decide to exit without accessing any worksheets, you can do so by pressing the **ATTN** key (or **SHIFT** **ATTN**). With the BUDGET worksheet in memory, you would see:

### Input/Result

Enter worksheet name: **BUDGET**

**ATTN**

> █

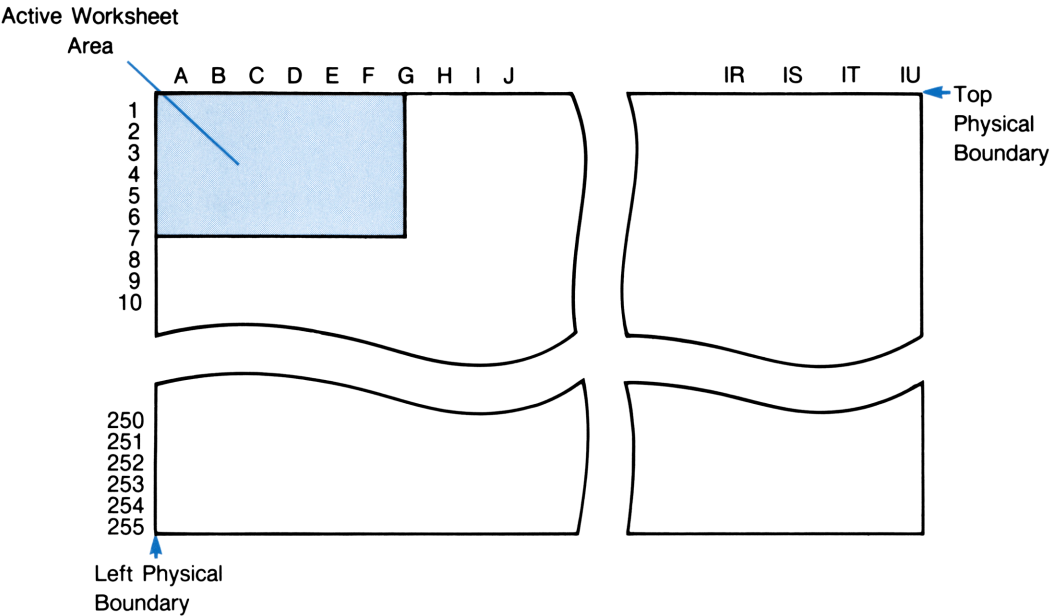
With the worksheet name template in the display, pressing the **ATTN** key exits the VisiCalc program and the prompt and cursor reappear.

Once you fully activate the VisiCalc program by specifying or entering a worksheet file name, you can only exit the program by pressing **SHIFT** **ATTN**.

## The VisiCalc® Display

While VisiCalc is active, your HP-75 display becomes a window into the worksheet file. The worksheet is organized into columns, which are lettered A, B, C, and so on, and rows, which are numbered 1, 2, 3, and so on. The intersection of each column and row specifies an entry position, or *cell*, with coordinates such as A1, B3, C17, etc. Into each cell position you can enter a message or title, a number, or a formula of the kind you might enter, keystroke by keystroke, on a calculator.

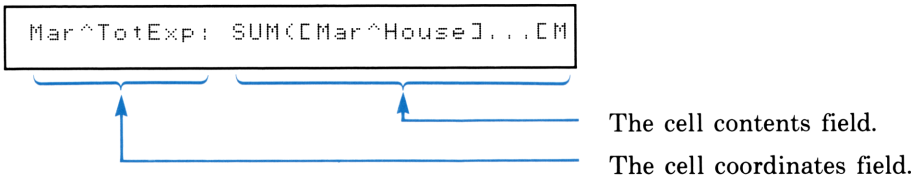
The illustration below shows the structure of the worksheet. In that illustration, the *active worksheet area* is the part of the worksheet that contains cell entries. It is bounded on the left by the *left physical boundary* of the worksheet and on the top by the *top physical boundary*. It is bounded on the right and on the bottom by the right-most column and the bottom-most row that contain nonempty cells. As you will see, certain cell-display movement keystrokes can be used to go directly to the active worksheet boundaries.



Maximum worksheet sizes and worksheet memory requirements are discussed in more detail later in this section under “Memory Management”(page 47).

### The Cell Display

When you first access a worksheet you will see the coordinates and contents of a single cell. This single *cell display* is the basic element of the worksheet that you will work with to build, examine, and modify the worksheet. Let’s look more closely at the contents of this display. To look at a typical cell display from BUDGET, press **[SHIFT] [>]**, then type **c9** and press the **[RTN]** key followed by the **[APPT]** key to set Formula Display mode. The cell shown below will be displayed.



The *cell coordinates* field gives the location of the cell within the worksheet. In this example the cell display is located at the intersection of the **Mar** column and the **TotExp** row. The cell coordinates field can contain *default* coordinates like **c9**, *user-defined* coordinates like those in the example above, or a mixture of both default and user-defined coordinates, like **[Mar]9**.

The *cell contents* field displays the cell entry. A cell can contain a numeric *value*, an alphanumeric *label*, or a *formula*. You can view the contents of a cell in either *Formula Display mode*—as in the illustration above—or in *Result Display mode*. Cells that contain labels or numbers look the same in either mode. Cell display modes will be described in detail in section 2 (page 61). A bit later we will discuss cell entry and editing in more detail (page 38).

The colon that separates the cell coordinates field from the cell contents field has a special function. When the colon is underlined, it indicates that the *value* shown in the cell you are viewing has a *formula* associated with it. When you are viewing the cell in Result Display mode, you can tell if there is a formula associated with the displayed value by the presence or absence of the underline.

Just as the cell *display* is a window into the entire *worksheet*, the cell *contents field* is a window into the *cell entry*. Since the length of a cell entry may exceed the size of that window, you need to be able to move the window back and forth so you can examine the hidden part of the entry. The **RUN** key and the **SHIFT** **RUN** keystroke combination enable you to do this.

- To move the *window* to the right one character position (and the *contents* to the left), press **RUN**.
- To move the window back to the left, press **SHIFT** **RUN**.

Try scrolling the current cell entry back and forth to see how these keys work.

Notice that the cell entry window scrolls “under” the cell coordinates field, and that the cell coordinates field never moves. The cell coordinate field stays fixed in the display, so you can always identify the current cell. Notice also that we didn’t use the left and right arrow keys (**←** and **→**) to scroll within the cell display. This is because the arrow keys are used to move around the *worksheet*—that is, from one cell to another. We will discuss further how to move around the worksheet shortly under “Accessing Cells” (page 32).

There are two other keystroke combinations that can be used to move the cell entry window, the **TAB** and **SHIFT** **TAB** keys.

- Pressing the **TAB** key moves the entry window one window width to the right or to the end of the entry string if less than a window-full of characters remain.
- Pressing the **SHIFT** **TAB** keystroke works the same, except that it moves the window to the left.

Try the **TAB** and **SHIFT** **TAB** keys on the cell in the display.

## Headers

When a worksheet is first created, all of the headers that define the cell coordinates in that worksheet are *default* headers. While the default headers do specify the column-row position of the cells in the worksheet, they provide little useful information about the relationship between the currently displayed cell and the total worksheet.

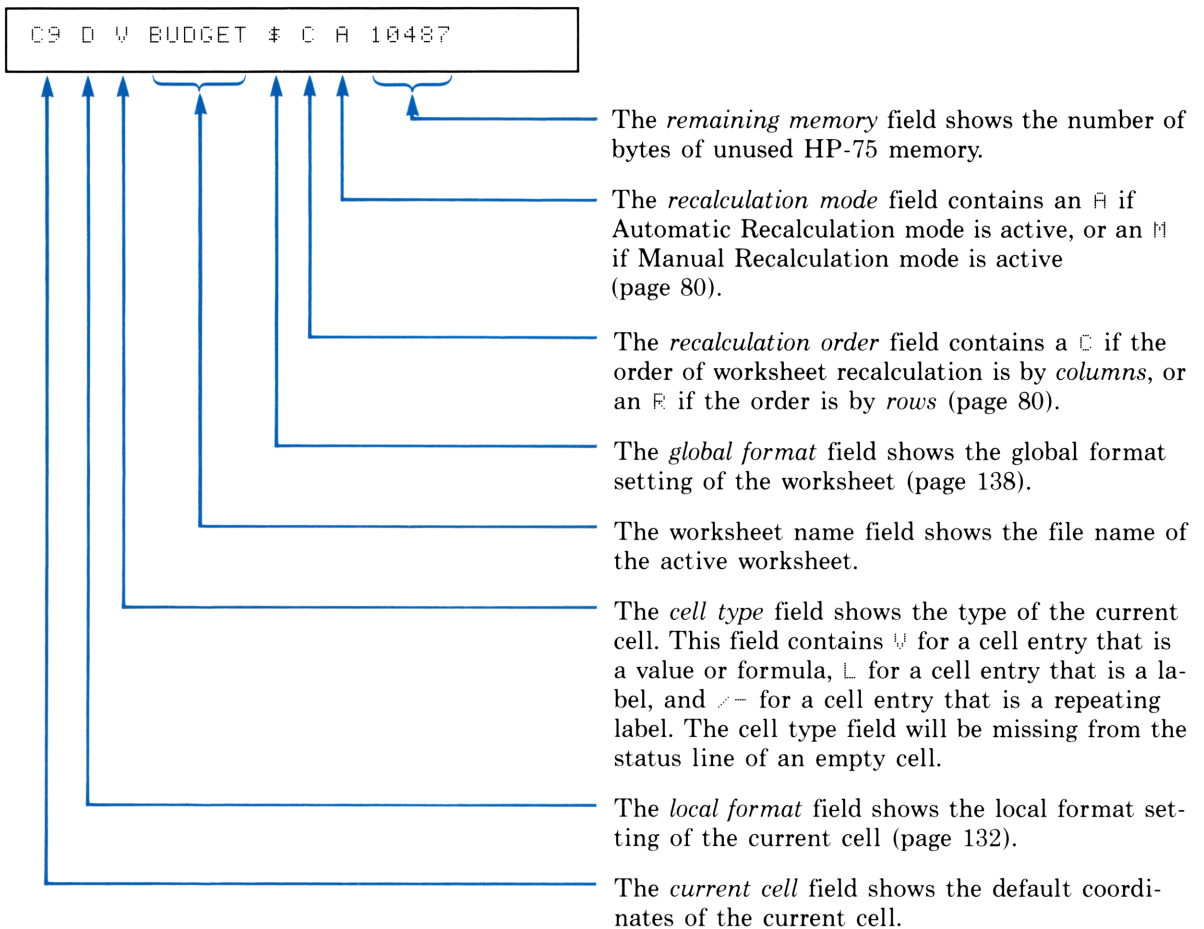
As you have already seen, VisiCalc allows you to create alphanumeric *user-defined* headers for columns and rows, and to use them to specify the position of a cell. This means that you can replace default column coordinates (like A, B, C, etc.) and row coordinates (like 1, 2, 3, etc.) with labels that describe the contents of the cell. Cell coordinates that describe the cell contents, or are otherwise logically related to the cell contents, are much easier to remember and use than coordinates like B5 or DR12. By defining and using these user-defined headers, you can better visualize the overall structure of the worksheet. User-defined headers can be comprised of up to 95 alphanumeric characters, including any character except ^, [, ], or !. You create them by using the /H (*header*) command (which is described in section 2, beginning on page 54). User-defined headers can be used anywhere that default headers are used—that is, any time that you need to specify the location of a cell.

When you specify the location of a cell with user-defined coordinates, they must be surrounded by the left and right square brackets ([ and ]), and the column and row headers must be separated by the caret symbol (^). For example, to enter the coordinates of a cell with column coordinate Jan and row coordinate Income, you would type [Jan^Income]. The brackets and caret are required so that VisiCalc can tell the difference between a user-defined and default header, and also between a numeric value and a header that consists entirely of numbers. For example, the user-defined coordinate [C^9] would look like default coordinate C9 without the brackets and caret.

## The Status Line

The *status line* contains certain information about the current cell and the active worksheet. It can be called into the display from Top level by pressing the [SHIFT][EDIT] keystroke combination. The status line contains several fields of information: some of these fields refer to the *current cell* (the cell that was in the display before the status line was called), others refer to the worksheet, and one refers to the computer's memory. Each of these fields is described below. Press [SHIFT][EDIT] to display the status line for the currently active worksheet:





To return to the current cell display, press **ATTN** or **SHIFT****EDIT**. The status line disappears and the current cell returns to the display.

### Input/Result

**ATTN**






Mar^TotExp: SUM(CMar^House],...CM


The current cell returns to the display.

## Accessing Cells

You'll need to be able to move the cell display around within the worksheet to examine, change, or add cells. VisiCalc provides three ways for you to accomplish this: the arrow keys, the Go To command, and the alternate viewing window.

### Using the Arrow Keys

You can use the four arrow keys (, , , ) and the arrow keys in combination with the  key to move to any cell in the worksheet. Pressing one of the arrow keys moves the cell display, one cell at a time, in the direction of the arrow.

**Example:** Press the right arrow key, , once:

#### Input/Result


```
Mar^TotExp: SUM([Mar^House]...[M
```















```
Apr^TotExp: SUM([Apr^House]...[A
```

The cell display moves one place to the right, to the column headed by **Apr**, and remains in the **TotExp** row.

Holding an arrow key down causes repeated moves in the same direction.

You can move the cell display to one of the boundaries of the worksheet in one step by pressing the  key with one of the arrow keys:

- The   and   keystrokes move the cell display to the top and left physical boundaries of the worksheet, respectively. If you press   then  , the cell display will move to cell **A1**—the upper-left corner of the worksheet.
- The   and   keystrokes move the cell display to the right-most active column and bottom-most active row of the worksheet, respectively.

Remember that the active boundaries delimit the area of the worksheet that contains nonempty cells.

Note that the unshifted arrow keys allow you to move past the active boundaries of the worksheet—all the way to the physical boundaries.



## Using the Go To Command

Another way of moving the cell display around the worksheet is to use the VisiCalc *Go To* command. This command moves the cell display directly to any cell in the worksheet from any other. To use the Go To command, press **[SHIFT][>]**, then enter the coordinates of the destination cell.

**Example:** Move the cell display to the cell with coordinates Jun^Income. First, press **[SHIFT][>]**—the display will then prompt for the coordinates of the destination cell. The destination cell can be specified by default, mixed, or user-defined coordinates.

### Input/Result

**[SHIFT][>]**

Go to: ■

[Jun^income] **[RTN]**

Jun^Income: [May^Income]

The display prompts for the coordinates of the desired cell; in this case, Jun^Income.

After you enter the coordinates of the destination cell, the desired cell is displayed. Note that you must surround user-defined coordinates with the left and right square brackets, and that you may enter the individual headers without regard to upper- or lower-case letters.

Now use default headers to go to cell Jan^Income:

### Input/Result

**[SHIFT][>]**

Go to: ■

a1 **[RTN]**

Jan^Income: 3000.00

VisiCalc again prompts for the coordinates of the destination cell.

Even though you used default coordinates, VisiCalc displays the user-defined coordinates if they exist for that cell.

You can use the Go To command to move the cell display to a different row in the current column or to a different column in the current row by specifying only a row or column header in the Go To command.

**Example:** To move to row 4 of the current column, execute the Go To command with only the row coordinate specified.

### Input/Result

SHIFT >

Go to: ■

4 RTN

Jan^Insur: 100.00

Enter only the row header (default or user-defined) of the destination row.

The cell display stays in the the same column (Jan) but moves to the 4th row (Insur).

Moves within a particular *row* can be accomplished in the same manner. Just specify the desired column when prompted for the cell coordinates.

## The Alternate Viewing Window

VisiCalc provides the means to quickly alternate your view of the worksheet between two separate cells or groups of cells. It does this by allowing you to create two “windows” into the worksheet—the primary viewing window that we have been discussing until now, and an *alternate viewing window*. You can alternate your view between the primary window and the alternate window with the press of a single key. Once you have switched to the alternate window, you can perform *any* VisiCalc operation—then return to the current cell in the primary window by again pressing a single key.

**Defining the Window Access Cell.** To create an alternate window you must first designate a window access cell. Once defined, this cell provides an access point to the alternate window. To do this, move the cell display to the desired cell, then press SHIFT FET. The cell display will blink once to indicate that VisiCalc is setting this cell as the window access cell. Note that you can perform this operation while the display is in either window. In a new worksheet, the window access cell is predefined to be cell A1.

**Accessing the Alternate Window.** To switch the display to this access cell in the alternate window, press the FET key. VisiCalc will immediately respond by switching the display from the current cell in the primary window to the window access cell in the alternate window. While the display is in the alternate window, VisiCalc underlines the coordinates field of the cell display.

The FET key is a toggle like TIME or APPT. When the display is in the alternate window (the coordinates field of the cell display is underlined), pressing FET will return the display to the current cell in the primary display.

**Windows As Reference Points.** It is important to realize that the creation of an alternate window does not separate the worksheet into two parts. Every cell in the worksheet is accessible from *either*

window. When you create an alternate window, you are actually defining two rapidly accessible reference points—the current cell in the primary window and the window access cell in the alternate window. The location of the reference point in the primary window changes constantly as you use the arrow keys or Go To command to move around the primary window. The location of the reference point in the alternate window is fixed until you deliberately define another cell as the access cell. If you press **[FET]** from the primary window, VisiCalc will display the window access cell—regardless of where the cell display was when you last left the alternate window.

Since you can perform any VisiCalc operation from the alternate window—including making and modifying cell entries, executing commands, even redefining the window access cell, you can take advantage of the different behavior of the reference points to increase the efficiency and flexibility of your worksheet operations. For example, you could make changes to a cell or cells in the primary window, then rapidly view the results of those changes in another cell or group of cells by defining the cell where the results occur as the alternate window access cell.

**Example:** Suppose, in the BUDGET worksheet, you want to see the effect of various income amounts in Jan^Income on the Remain (remainder after expenses) row—starting with Jan^Remain. To do this you could designate Jan^Remain as the alternate window access cell, then go back to Jan^Income and make a change. To observe the effect of the change you would simply press **[FET]** to switch to Jan^Remain in the alternate window, then use the right arrow key to scan along the Remain row. To make another change you could press **[FET]** to return to Jan^Income, make the change, then press **[FET]** again to return to Jan^Remain and repeat the process.

### Input/Result

**[SHIFT] > [Jan^remain] [RTN]**

Moves the cell display to Jan^Remain.

Jan^Remain: [Jan^Income]-[Jan^To

**[APPT]**

Switches the display to Result Display mode.

Jan^Remain\_ 1090.00

**[SHIFT] [FET]**

Designates Jan^Remain as the alternate window access cell. The display blinks once to indicate that the cell has been designated.

Jan^Remain\_ 1090.00

SHIFT ↑

Jan^Income: 3000.00

Returns to Jan^Income.

2500 RTN

Places a new value in Jan^Income.

Working  
Jan^Income: 2500.00

FET

Switches to the alternate window. Note that Jan^Remain has changed.

Jan^Remain: 840.00

→

Moves the alternate window cell display to Feb^Remain. The rest of the Remain row could be examined in turn.

Feb^Remain: 880.00

FET

Returns to Jan^Income for another change.

Jan^Income: 2500.00

2750 RTN

Makes another change to Jan^Income.

Working  
Jan^Income: 2750.00

FET

Switches back to the alternate window. Note that the display returned to Jan^Remain, not to the last cell viewed in the alternate window.

Jan^Remain: 965.00

The example above illustrates just one of the possible ways that the alternate viewing window can be used to speed up your work. By remembering the differing natures of the two reference points described above, you will easily find other situations in which an alternate window can be helpful.

## VisiCalc® Operational Levels





When you first enter the VisiCalc program, VisiCalc is set to *Top level*; from this level you can enter two other VisiCalc levels: *Edit Level* and *Command level*. The operation of some of the computer keys differs at different levels, and in most cases you must return to Top level before selecting another level. We will discuss Edit level in this section and in more detail in section 2, “Creating and Editing Worksheets” (page 53). We will also briefly examine Command level in this section.

In practice you will seldom need to worry about VisiCalc levels. VisiCalc automatically switches between levels as required, leaving you free to enter data, execute commands, or perform editing. When you signal to VisiCalc that you are through with an operation (by terminating an entry with the **RTN** key, “backing out” of a command with the **ATTN** key, etc.), VisiCalc automatically returns to Top level, ready for a new entry or command. In almost any situation, you can return to Top level by simply pressing **ATTN**. In fact, the principal use of the **ATTN** key is to stop an operation before it is completed and return VisiCalc to the condition it was in before the operation was begun.

## Top Level

The principal activity at Top level is viewing the worksheet. You have already seen how to view the worksheet using the arrow keys, the Go To command, and the alternate window. You have also seen how to examine the contents of a particular cell using the **RUN**, **SHIFT** **RUN**, **TAB**, and **SHIFT** **TAB** keys. There are several other operations, most related to viewing the worksheet, that you can carry out at Top level. These operations are each performed by a single key or keystroke combination. The list below summarizes the operations that can be performed at Top level. Some of these operations have already been described in this section. The rest are listed here to introduce them, but they will be covered in more detail later in the manual.

Key	Operation
<b>ATTN</b>	Returns to Top level.
<b>SHIFT</b> <b>ATTN</b>	Exits VisiCalc.
<b>TIME</b>	Toggles the cell coordinates field of the cell display between User-Defined and Default Coordinate Display modes. (Section 2, page 62.)
<b>SHIFT</b> <b>TIME</b>	Toggles the cell contents field of the cell display between User-Defined and Default Coordinate Display modes. (Section 2, page 62.)
<b>APPT</b>	Toggles the worksheet between Formula and Result Display modes. (Section 2, page 62.)
<b>SHIFT</b> <b>APPT</b>	Toggles the cell display between Formatted and Unformatted Display modes. (Section 2, page 63.)
<b>SHIFT</b> <b>EDIT</b>	Toggles the display between the current cell and the worksheet status line.
<b>FET</b>	Toggles the display between the current cell and the access cell of the alternate window.
<b>SHIFT</b> <b>FET</b>	Defines the access cell of the alternate window.
<b>↑</b> , <b>↓</b> , <b>←</b> , <b>→</b>	Moves the cell display one cell up, down, left, and right, respectively.

Key	Operation
<b>SHIFT</b>  ,	
<b>SHIFT</b>  ,	
<b>SHIFT</b>  ,	
<b>SHIFT</b>  ,	Moves the cell display to the top and left physical boundaries, and to the bottom and right active boundaries of the worksheet, respectively.
<b>RUN</b>	Moves the cell contents of the cell display one character position to the left.
<b>SHIFT</b> <b>RUN</b>	Moves the cell contents of the cell display one character position to the right.
<b>TAB</b>	Moves the cell contents of the cell display one window width to the left.
<b>SHIFT</b> <b>TAB</b>	Moves the cell contents of the cell display one window width to the right.
<b>!</b>	Recalculates the entire worksheet. (Section 3, page 80.)
<b>@</b> , <b>+</b> , <b>-</b> , <b>.</b> , <b>(</b> , <b>)</b> , or digit	Switches VisiCalc to Edit level, ready for a value entry.
<b>'</b> , <b>"</b> , <b>^</b> , <b>#</b> , <b>\$</b> , <b>%</b> , <b>&amp;</b> , letter, or space	Switches VisiCalc to Edit level, ready for a label entry.
<b>EDIT</b>	Switches VisiCalc to Edit level, ready to change the current cell contents. (Section 2, page 58.)
<b>CLR</b>	Clears the contents of the current cell and switches VisiCalc to Edit level, ready for a new entry of the same type as the just-cleared entry.
<b>SHIFT</b> <b>&gt;</b>	Initiates the Go To command.
<b>/</b>	Switches VisiCalc to Command level. (Section 1, page 46.)

## Edit Level

To “write” on a VisiCalc worksheet, you simply move the cell display to the desired location, then begin keying in the cell contents from the keyboard. As you begin typing, VisiCalc immediately switches to *Edit* level. When VisiCalc switches to Edit level in this manner, the current cell contents—if any—are overwritten by the new entry when you press **RTN**. However, if you press **ATTN** to return to Top level *without pressing RTN*, the current cell contents are preserved and the new entry is lost. This is because VisiCalc keeps the new cell entry in the display buffer until the **RTN** key is pressed—only then does it store the new entry in the cell and replace the existing cell contents.

In Edit level all of the HP-75 editing keys perform their normal function. Before pressing the **RTN** key to terminate cell entry, you can use the editing keys to correct typing mistakes or change the text of a



cell entry. In fact, the HP-75 editing keys are active whenever the blinking cursor (■ or ✱) is present in the display. It is important to remember this since many of the HP-75 editing keys perform different operations at other VisiCalc levels.

You can also switch from Top level to Edit level by pressing the **[EDIT]** key. When you do this, VisiCalc switches to Edit level and the current cell contents are made available for editing. The two ways of entering Edit level and of editing cell entries are described in detail in section 2, beginning on page 57.

## Labels

Labels are generally used to title or explain the information and calculations on the VisiCalc worksheet. VisiCalc interprets a cell entry as a label if the first character keyed in is an upper- or lower-case alphabetic character (A through Z, or a through z), a single or double quote ( ' or " ), ^, #, \$, %, &, or the space bar. Label entries can contain up to 95 characters.

*VisiCalc looks at only the first character of a cell entry to determine whether the entry is a label or a value; after the first character has set label mode, you can use any character accessible from the keyboard as part of a label.*

**Note:** If you use characters with ASCII\* decimal codes less than 32 or greater than 127 in cell entries (or headers), you may get unexpected results when those characters are displayed on a video display device or printed. Some of the character codes in these ranges are control codes that affect the operation of video displays and printers.

Let's start with a new, blank worksheet and make some entries to see how labels work. To do this you will need to exit VisiCalc, then reactivate the system and create a new worksheet:

### Input/Result

**[SHIFT]** **[ATTN]**

```
Leaving worksheet: BUDGET
>■
```

vc **[RTN]**

```
VisiCalc (R) - copyright HP 1983
Enter worksheet name: BUDGET
```

**[CLR]** sales **[RTN]**

When you exit the VisiCalc program, it displays the exit message for one-half second, then the HP-75 EDIT prompt and cursor are displayed.

Clears the name field and enters the new worksheet's file name.

---

\* ASCII stands for American Standard Code for Information Interchange.



A1:

VisiCalc displays the first cell of the new worksheet.

Now to place the label `Sales` in cell A1, you simply type the word and terminate the entry by pressing `RTN`. As you type the label, the HP-75 will beep when you enter a character at the position in the entry that corresponds to the current *column width*\* of the column in which the cell resides. You aren't limited to that many characters, however. In a newly created worksheet, the width of all the columns in the worksheet is preset to seven characters. Let's enter the label `Sales` and see what happens to the display. Key in the label and press `RTN`:

### Input/Result

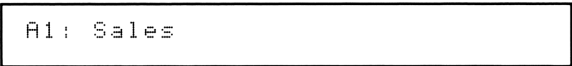
`Sales`



A1(L): Sales■

As soon as you start typing the label, the cursor appears (indicating that VisiCalc has switched to Edit level), and the `(L)` appears (indicating that this is a label entry).

`RTN`



A1: Sales

After you press the `RTN` key, the cursor disappears (indicating that VisiCalc has returned to Top level), and the label indicator also disappears.

Now cell A1 contains the label `Sales` and VisiCalc is ready for you to move to another cell or enter a command.

If you attempt to type in a label but precede it with a value character, such as `-`, `[`, or a digit, VisiCalc assumes that you are entering a value, and the label will not be accepted by the computer. Using the single or double quote symbols (`'` or `"`) as the first character keyed in tells VisiCalc that the expression is a label, no matter *what* else follows. (The quote symbol does not appear in the final expression, and no closing quote symbol is needed.)

For example, let's try to type a label for cost at the beginning of the second row—in cell position A2, but we'll precede the label with a dash.

### Input/Result

`[-Cost-`

Pressing `↓` moves the cell display to cell A2 so you can key the entry into the display.



A2(V): [-Cost-■

The `(V)` indicates that VisiCalc thinks the entry is a *value* entry.

\* Column width and setting the column width are both discussed in section 8, "Formatting the Worksheet" (page 131).



RTN

A2(V): -Cost-

Pressing **RTN** terminates the entry; the HP-75 beeps to indicate an error, and VisiCalc will not accept the entry. VisiCalc returns the cursor to the first invalid character of the entry—the **V** in this case—to prompt you to make corrections.

Since we want this to be a *label* entry, not a *value* entry, we must reset the cell to the blank condition to clear the error, then type the correct entry. To do this, press **ATTN** to cancel the entry and return to Top level, then retype the label beginning with a single or double quote.

### Input Result

ATTN

Clears the invalid entry.

A2:

" -Cost-

Enters a valid label.

A2(L): -Cost-■

RTN

Now when you press **RTN** to terminate cell entry, VisiCalc accepts the entry as a label because the double quote, **"**, told VisiCalc that you were typing a label.

A2: -Cost-

## Values

Values may be thought of as the actual numeric entries on the VisiCalc worksheet. In general however, a value entry is an expression (or formula) consisting of any combination of numbers, arithmetic operators (such as + and \*), parentheses, functions, or cell references (that is, cell coordinates that refer to other cells). Value entries can contain up to 91 characters.

**Example:** Let's write a value for *Sales* in column B. Move the cell display to cell B1 and enter a value representing \$100:

### Input/Result

SHIFT &gt; B1 RTN

Moves the cell display to cell B1.

B1:

100

B1(V): 100■

RTN

Working  
B1: 100

As soon as you press the first numeric key, VisiCalc immediately knows that the entry is to be a value.

The **RTN** key terminates entry and stores the value in the cell.

Formulas

The real power of VisiCalc comes from its ability to have the contents of one cell calculated using the contents of another—or several others. By using, as a cell entry, a formula that depends on the value in another cell, and that cell on another cell, etc., you can play “what if,” changing the entire worksheet by altering only a few values.

To use the value of one cell as part of a formula in another, simply include the cell’s coordinates in the formula in the second cell. You can use default, user-defined, or mixed coordinates to specify a cell in a formula. For a cell entry to be a calculated value, it cannot begin with an alphabetic character or the quote symbol. (A reference to an empty cell or to a cell that contains a *label* will be evaluated as zero.)

**Example:** There are two rows in our SALES worksheet, Sales and -Cost-. We have a value of \$100 in the Sales row, and we want the entry for -Cost- at cell B2 to be 60% of the value for Sales given in cell B1( $B2 = .6 \times B1$ ). To accomplish this we will use B1 as part of the formula in B2.

Input/Result

↓ .6\*b1

B2(V): .6\*b1■

RTN

B2: 60

Moves to cell B2 and enters the formula in the cell display.

The **RTN** key terminates cell entry.

VisiCalc calculates the *result* of the formula you entered and displays that result, 60, in the cell display.

But while the display shows the result of the calculation, the formula is also stored in cell B2. You can see this by pressing **APPT** to toggle between Result and Formula Display modes.

**Input/Result**

APPT

B2: .6\*B1

Formula Display mode causes any formula associated with a cell to be displayed instead of the result of the formula.

If you change the value in cell B1 now, the contents of B2 will also change, since the formula in B2 includes a reference to cell B1.

**Example:** Let's change the `Sales` amount in cell B1 to \$200.

**Input/Result**

APPT

Changes the cell display back to Result Display mode.

↑ 200 RTN

Enters the new sales amount in cell B1.

B1: 200

↓

Moves to cell B2 to see the result of the change.

B2: 120

When you changed the value in cell B1, VisiCalc recalculated the value of cell B2. `-Cost-` is still 60% of `Sales`, but the *value* of `-Cost-` has changed because the value of `Sales` has changed.

At this point the worksheet looks like this:

SALES:	A	B
1	Sales	200
2	-Cost-	120

Now let's create a third row that will calculate the difference of `Sales` and `-Cost-`. First, give the row the label `Gross`.

**Input/Result**

SHIFT &gt; a3 RTN Gross RTN

Enters the new label in A3.

A3: Gross

Now, for the value of `Gross` that will appear in cell `B3`, we need to calculate `Sales` minus `-Cost-`; that is, `B1-B2`.

But remember how VisiCalc determines whether you are typing a label or a value. It looks at the first character typed—and the first character of `B1-B2` is an alphabetic character, `B`. `B1-B2` is a perfectly legitimate label, but it doesn't calculate anything. If you typed `b1-b2` into cell `B3`, you would not have the result of calculating `Sales` minus `-Cost-`, but simply a label.

Remember that we used the single or double quote keys (`'` and `"`) to tell VisiCalc that the following characters are to be treated as a label. We can use a similar technique to type the formula `B1-B2` as a value, instead of as a label. The simplest way to do this is to put a plus sign (+) before the first cell reference. VisiCalc will then interpret the entry as a value entry.

**Input/Result**

`→` `+b1-b2`

`B3(V): +b1-b2`

`RTN`

`B3: 80`

Pressing `→` moves to cell `B3` so you can enter the formula. Notice that the `(V)` appears as you type the `+`, indicating that this is a value entry.

When you press `RTN`, VisiCalc calculates the result and displays it.

After the above operations are completed, the worksheet should look like this:

SALES:	A	B
	:	
	-----+	-----
1	!Sales	200
2	!-Cost-	120
3	!Gross	80

The `@` key can also be used to force VisiCalc to interpret an entry as a value entry. The `@` key is also unique, in that you can convert a label entry to a value entry *after* the entry is started. You do this by inserting an `@` character at the beginning of the entry (use the arrow keys to move the cursor to the beginning of the entry, press the `I/R` key to display the insert cursor (`*`), then press the `@` key). When you press `RTN`, the entry is converted to a value entry and stored.

## Functions

Formulas can contain calls to functions, as well as values, cell references, and operators. Function *calls* consist of the function name followed by a *parameter list* in parentheses. Listed below are the functions built in to VisiCalc:

AVERAGE	Computes the arithmetic mean or average of the numeric parameters in a list.
ERROR	Results in an “error” value that makes all expressions using the value display ERROR in the cell display.
MAXL	Computes the maximum value of the parameters in a list.
MEAN	Computes the arithmetic mean. MEAN is another name for the AVERAGE function.
MINL	Computes the minimum value of the parameters in a list.
NA	Results in a “not available” value that makes all expressions using the value display NA. Allows the effect of a particular value to be traced through the worksheet.
SUM	Computes the sum of the parameters in a list.

These functions and the terms *parameter* and *list* are described in more detail in section 3, “Using Functions and Formulas” (page 69).

In addition to the built-in functions listed above, VisiCalc allows you to use any HP-75 numeric function that takes only numeric parameters and returns a single numeric value. The HP-75 functions that can be used with VisiCalc are listed in appendix B, “VisiCalc Functions” (page 175).

## Extension Functions

Another powerful feature of VisiCalc is its ability to access functions that are external to the VisiCalc system. These “extension functions” are specially designed BASIC programs and can be included in a formula just like HP-75 and regular VisiCalc functions. By using extension functions, you can extend the function set of VisiCalc to include almost any numeric computation you might need.

Extension function calls look exactly like built-in function calls. The only difference is that the function name is that of a BASIC program file that *you* create. Extension functions are fully described in section 10, “Extending the VisiCalc Function Set” (page 151).

A magnetic card containing an extension function named IRR (internal rate of return) is packaged with VisiCalc. Although IRR was included primarily to help illustrate the information on VisiCalc extension functions in section 10, you may wish to use it to solve IRR problems. The use of IRR to solve internal rate of return problems is described in section 10 (page 152).


## Accessing Other Worksheets in Memory

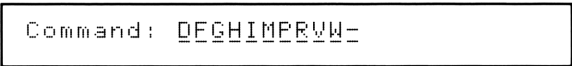
Another type of cell entry that you can use is an *external cell reference*, which is a reference to another worksheet in memory. External cell references can reference a single cell or a range of cells, and they can be used in expressions, formulas, and function parameter lists.

External cell references are described in detail in section 9, “Accessing Another Worksheet From a Cell” (page 143).

## Command Level

VisiCalc has 11 commands to help you control input, worksheet and number format, worksheet calculations, user-defined headers, video display of the worksheet, and printed output.

To select a VisiCalc command, you first enter Command level by pressing the slash character key, . VisiCalc then prompts:



The VisiCalc command menu.

Each underscored character is the first character of a VisiCalc command. To select a command, you press the key that corresponds to the first letter of the desired command. Some commands have their own sets of additional subcommands to enable you to control or to select their options.

The operation of each command is covered in detail elsewhere in this manual. Here is the VisiCalc command set:

Command	Operation
/D	<i>Delete.</i> Deletes the column or row in which the cell display resides. (Section 2, page 66.)
/F	<i>Format.</i> Sets the display format of the current cell display. (Section 8, page 132.)
/G	<i>Global.</i> Sets global status for column width and cell format, sets Automatic or Manual Recalculation mode, and sets the recalculation order to column or row. (Section 8, page 138.)
/H	<i>Header.</i> Sets the display width of user-defined row headers, allows you to create and edit user-defined column or row headers, and allows you to suppress the printing or video display of default headers. (Section 2, page 54.)
/I	<i>Insert.</i> Inserts a blank column or row. (Section 2, page 64.)
/M	<i>Move.</i> Moves a column or row within the worksheet. (Section 2, page 67.)
/P	<i>Print.</i> Used to print the worksheet on an external printer. (Section 6, page 105.)
/R	<i>Replicate.</i> Allows you to repeat a cell entry or a group of cell entries across a column or row or a group of columns or rows. (Section 4, page 85.)

Command	Operation
<b>/V</b>	<i>Video.</i> Allows you to activate or deactivate the video display. (Section 7, page 120.)
<b>/W</b>	<i>Width.</i> Sets a local column width. (Section 8, page 136.)
<b>/-</b>	<i>Repeating Labels.</i> Allows you to fill a cell with repetitions of a single character or group of characters. (Section 2, page 61.)

## Exiting the VisiCalc Program

You use the **[SHIFT][ATTN]** keystroke combination to exit the VisiCalc program at any time—from any level. When you press **[SHIFT][ATTN]**, VisiCalc responds:

Leaving worksheet: *worksheet name*\*

Then the HP-75 EDIT prompt and cursor reappear. Remember that your worksheet file is still in continuous memory and can be accessed again by restarting the VisiCalc program.

## Returning to Top Level

After any command completes execution or you complete a cell entry, VisiCalc automatically returns to Top level. With one exception, you can also return to Top level any time by pressing the **[ATTN]** key. The single exception occurs when entering user-defined column or row headers. Pressing **[ATTN]** while a header is being entered causes the entry to be aborted and the header coordinates to be redisplayed so you can enter another header. A second press of the **[ATTN]** key at this point will cause VisiCalc to return to Top level.

## Memory Management

Since the VisiCalc program is stored in a plug-in module, no user memory is used to store the program itself. However, VisiCalc uses approximately 2500 bytes of RAM whenever it is active and certain operations use an additional amount of RAM that varies with the operation. The system overhead requirements of VisiCalc and of the HP-75 combined leave approximately 12 kilobytes (20 kilobytes with the HP 82700A Memory Module installed) of user memory available for storing worksheets and other files. When you create worksheet files, they require an amount of RAM that depends on the number of active cells in the worksheet.

## Dynamic Worksheet Size

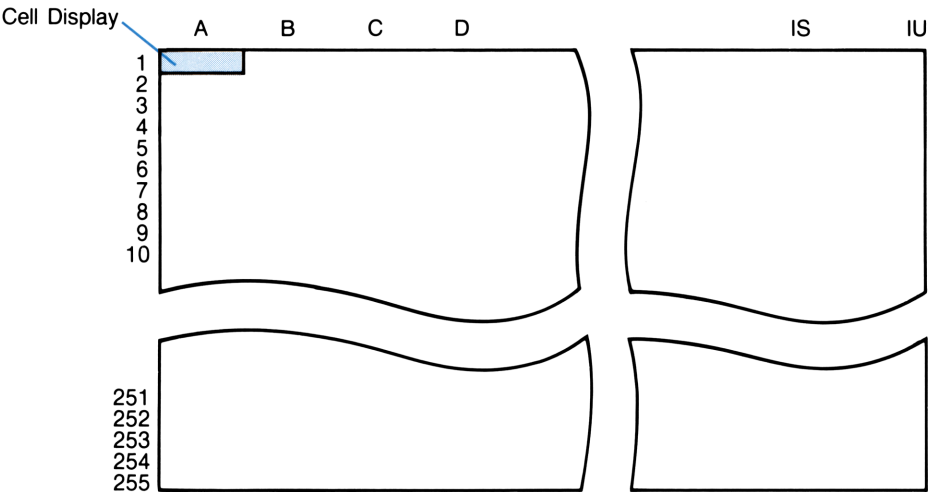
As you edit and modify a worksheet, the amount of memory used to store the worksheet file grows and shrinks as the size of the worksheet changes. The basic philosophy behind worksheet memory management is “return every possible byte to the user.” Whenever an operation causes the worksheet to need

---

\* The *worksheet name* that appears in this display is, of course, the HP-75 file name of the worksheet.



less memory space, VisiCalc automatically shrinks the worksheet file and returns the unused bytes to free memory. Conversely, as you add cells to the worksheet, VisiCalc automatically allocates memory for those cells and expands the worksheet file to accommodate them.



There are 255 columns in the worksheet, labeled A through Z, AA through AZ, BA through BZ, and so on to IA through IU. There are also 255 rows, so VisiCalc allows a total of  $255^2$  or 65,025 possible cells. Since each cell uses several bytes of user memory, the actual limit is less and is constrained by available memory.

A newly created, but empty, worksheet file requires 22 bytes of memory. Memory bytes are used by cells at the rate of 2 bytes per cell position. So if you create a large worksheet, but leave most of the cells blank, the blank cells will each use 2 bytes of memory, regardless of whether the cells contain any information. As you write labels or values into a cell, more memory is used for each additional character.

Even though VisiCalc lets you scroll down to cell IU255, the worksheet is not really this large. In fact, as mentioned above, the worksheet is dynamically reconfigured; it grows and shrinks to the size and shape you need as you use it. You actually start with a 1 by 1 worksheet, starting and ending at cell A1. Although you can move the cell display to any cell in the 255 by 255 matrix, no additional memory is allocated for the sheet until you actually write something on it. As you make cell entries the sheet grows into a rectangle just large enough to include the highest-numbered column and row that contain nonempty cells.\* As you move further to the right and down, writing labels, numbers, and formulas, this process continues until available memory is exhausted.

\* A cell is nonempty if it contains an entry or if it has a local format (page 132).

If you begin with a blank worksheet, then write an entry at cell I11, the sheet will grow to a size of 1 row by 255 columns. If you move the cell display to any cell in row 255 and try to write an entry there, the entry will not be accepted because you are attempting to extend the worksheet to a rectangle of 255 columns by 255 rows, and this requires far more memory than that available in your HP-75. Even with no other files in memory, the maximum number of cells that a worksheet can contain is less than 6000. (This assumes a standard 16K HP-75 and a completely empty worksheet.) For worksheets that contain information, this translates into about 10 worksheets the size of BUDGET, or one worksheet with about 600 cells.

The *remaining bytes* field of the status line will show the decrease in available memory as you write information on the worksheet (remember that you press **SHIFT** **EDIT** to display the status line), or as you use commands that write on the worksheet. If you exhaust available memory, VisiCalc will beep, the program will terminate, then error 16—not enough memory—will be displayed. Also, because certain VisiCalc operations use some temporary memory, an out-of-memory error may occur during those operations—even if the remaining memory field of the status line indicates that enough memory should be available. This can occur while making a cell entry, while inserting a column or row, during replication, during worksheet recalculation, etc. To recover from an out-of-memory condition, you must either reduce the size of the affected worksheet, or remove other files from memory.

## Multiple Files in Memory

In practice, you don't need to be too concerned about memory management since VisiCalc manages memory efficiently and automatically. However, as you fill the HP-75 memory with files—whether worksheet, BASIC program, text, or other files—you may need to purge unneeded files or copy them to mass storage to make room for new worksheet files. Also, as the size of any particular worksheet grows you will want to periodically examine the remaining memory field of the status line so you can take steps to make room for all of your entries.

Because you can have many files in memory at once, VisiCalc has been designed to use memory as efficiently as possible so you can take advantage of this capability. VisiCalc worksheet files are listed in the file catalog of the HP-75 as type W files. As mentioned earlier, they can be CATALOGUED, COPIED, PURGED, or RENAMED, but not LISTED, PLISTED, RUN, EDITED, CALLED, RENUMBERED, or TRANSFORMED.

## Purging Worksheet Files

To remove a worksheet from memory, you must exit VisiCalc via the **SHIFT** **ATTN** keystroke, then execute the HP-75 PURGE command.

**Example:** To purge the file that contains the SALES worksheet created earlier, first press **SHIFT** **ATTN** to exit VisiCalc.

### Input/Result

**SHIFT** **ATTN**

VisiCalc displays the program termination message when you exit the system.

```
Leaving worksheet: SALES
> █
```

purge 'sales' **RTN**

The PURGE command removes the the SALES file from memory.

```
> █
```

Further information on using the PURGE command can be found under “Purging Files,” in section 3 of the *HP-75 Owner's Manual*.

## VisiCalc® System Requirements

Together, your HP-75 and VisiCalc form a fully functional problem-solving system that can be used almost anywhere. However, certain HP-75 accessories or peripheral devices are needed to perform certain VisiCalc operations or to expand the capability of the system:

The following list details these expanded system requirements:

- To increase the memory capacity of the HP-75 so that more or larger worksheets can be maintained in memory, use the HP 82700A 8K Memory Module.
- To provide for more flexible and rapid mass storage of worksheets than that provided by the built-in HP-75 card reader, use an HP 82161A Digital Cassette Drive or any other HP-IL compatible mass storage device. (Mass storage operations with VisiCalc are discussed in section 5, page 103.)
- To produce printed copies of your worksheets, use either the HP 82162A Thermal Printer (24-character line and portable operation), the HP 82905B Opt 248 Impact Printer (full size, desk top, dot matrix printer), the HP 2671A Opt 048 Alphanumeric Printer, the HP 2671G Opt 048 Graphics Printer, or any other HP-IL compatible printer. (Printing worksheets is discussed in section 6, page 105.)
- To provide a larger viewing window into your worksheets, use an HP 82163 Video Interface and a video display device (video monitor or television set), or any other HP-IL compatible video interface and display device. (Using a video display with VisiCalc is discussed in section 7, page 119.)

All of the peripherals and accessories listed above can be used together or in any combination to provide a system that exactly fits your needs.

## Where Do We Go From Here?

By now, you have learned enough of the basics of VisiCalc to begin setting up solutions to your own applications. As you proceed further in your study, you will find that VisiCalc was designed with you, the user, in mind. As you become familiar with VisiCalc, you will discover that you seldom need to worry about how a particular operation or command works, since VisiCalc's friendly commands lead you through the most complex of operations.

There is still much more to VisiCalc than you've learned so far. In this section we've only touched on many of the operations and concepts that will be more fully described in later sections. In addition, in later sections you will learn how to use mass storage with VisiCalc, how to print your worksheets, how to use a video display with VisiCalc, how to arrange the worksheet format to fit your own needs, how to use external cell references, and how to use and create extension functions.

Of course, you can simply use the index and the table of contents when you need to learn something new. But we urge you to continue working through this manual, step-by-step, until you've seen all the functions and commands of VisiCalc. Your mastery of this powerful tool will be greatly enhanced for your efforts.



## Section 2

# Creating and Editing Worksheets

## Contents

What You'll Learn in This Section .....	53
Headers .....	54
Creating a Header .....	54
Editing a Header .....	57
Creating and Editing Cell Entries .....	57
Edit Level Operations .....	58
Editing Previously Entered Cell Contents .....	59
Repeating Labels .....	61
Using Redefined Keys .....	61
Modifying the Cell Display .....	61
Displaying a Formula or a Result .....	62
Changing Coordinate Display Modes .....	62
The Display Format Toggle .....	63
Column or Row Manipulations .....	64
Inserting a Column or Row .....	64
Deleting a Column or Row .....	66
Moving a Column or Row .....	67

## What You'll Learn in This Section

This section illustrates how to create and edit user-defined headers and how to create and edit cell entries. It also illustrates how to insert, delete or move a column or row of your worksheet. When you finish this section, you should be able to:

- Create and edit user-defined headers.
- Edit during cell entry.
- Edit a previously defined cell.
- Create *repeating labels*.
- Change the Coordinate Display modes.
- Switch between Unformatted and Formatted Display modes.

- Insert a column or row into a worksheet.
- Delete a column or row from a worksheet.
- Move a column or row to a different position in a worksheet.

Before you read this section, you should have read and understood the information in section 1.

## Headers

Remember from section 1 that default headers are specified with column and row coordinates (such as A, B, C, and 1, 2, 3, etc.), user-defined headers are specified with labels (such as [Jan^Income]), and mixed headers are specified with both user-defined and default headers (such as [Jan]2 or ACIncome).

User-defined headers can be any length up to 95 characters. However, the first 8 characters of every user-defined header must be unique—*duplicate user-defined headers are not allowed*. When you enter a user-defined header the HP-75 will beep when the eighth character is entered. When the header is subsequently viewed on the display, only the first 8 characters of the header will be shown and they will be enclosed in brackets ([ and ]) to indicate that this is the part of the header that must be used for cell references.\* The brackets are not part of the header, and should not be used when entering a user-defined header.

Any characters can be used in user-defined headers except ^, [, and ]. User-defined headers can be used any place that default headers can: in the Go To command, in formulas, to specify coordinates in commands such as Move and Replicate, etc.

## Creating a Header

The /H (Header) command is used to create user-defined headers. To do so:

1. Move the cell display to any cell in the column or row that you wish to define, then press [/] [H]. The following prompt will be displayed.  

```
Headers: Col Row Width Suppress
```
2. Do one of the following:
  - Press [C] to enter column headers. VisiCalc will prompt for the column header of the column where you are currently positioned.
  - Press [R] to enter row headers. VisiCalc will prompt for the row header of the row where you are currently positioned.

---

\* When a worksheet is displayed on a video display device or is printed, you can set the width of the worksheet columns so that all or any part of a header can be displayed. Refer to section 8, “Formatting the Worksheet” (page 131), for information on setting column widths.



- Press **[W]** to change the header width. VisiCalc will prompt for the row header width. When a worksheet is first created, the row header width is 5 characters. This command alters the appearance of video displays or printed worksheets.
- Press **[S]** to suppress the headers. This command suppresses default headers on the video display or the printer. VisiCalc will prompt:

Suppress default headers: Yes No

Press **Y** to suppress the default headers or **N** to restore the default headers.

**Example:** Create headers for the following worksheet, which we will call **COSTS**.

COSTS	Qtr 1	Qtr 2
Labor		
Material		
# Units		
Unit Cost		

Input/Result

vc **[RTN]**

Enters VisiCalc.

Enter worksheet name: **B**UDGET

Prompts for the new worksheet name.

**[CLR]** costs **[RTN]**

Names the new worksheet **COSTS**. After the worksheet is named, VisiCalc displays the first cell.

A1:

**[/]** **[H]**

Executes the Header command.

Headers: Col Row Width Suppress

VisiCalc displays the header options.

**[R]**

Selects the row header option.

Row header 1:

Labor **[RTN]**

Enters and stores the first row header. All of the characters that start label and value entry can be used to begin a header.\*

\* If you wish to begin entering a header with a character that does not start label or value entry, you must press **[^]** or **["]** first.

Row header 1: [Labor]



Row header 2:

Material [RTN]

Row header 2: [Material]



Row header 3:

# Units [RTN]

Row header 3: [# Units]



Row header 4:

Unit Cost [RTN]

Row header 4: [Unit Cos]

[ATTN]

ACLabor]:

[/] [H] [C]

Col header A:

VisiCalc displays the header of row 1.

Moves to the second row.

Enters and stores the second row header. The HP-75 will beep when the eighth character is keyed in.

Moves to the third row.

Enters and stores the third row header.

Moves to the fourth row.

Enters and stores the fourth row header. Notice that only the first 8 characters of the header are displayed.

Goes back to Top level and displays the cell where the /H command was executed. The first column does not have a user-defined header and defaults to A. The first row has header Labor.

Selects the column header option.

Qtr 1 **RTN**

Col header A: [Qtr 1]

Enters and stores the first column header.

→ Qtr 2 **RTN**

Col header B: [Qtr 2]

Moves to the second column and creates the second column header.

→ Ytd **RTN**

Col header C: [Ytd]

Moves to the third column and creates the third column header.

**ATTN**

Returns to Top level.

Qtr 1^Labor:

## Editing a Header

If you make a mistake while typing in a header (before **RTN** is pressed), the editing keys (which we will discuss next in “Creating and Editing Cell Entries”) can be used to correct your entry. Or, you can press **ATTN** to cancel the current header entry and return to the previous entry.

If you wish to change a header *after* you have stored it (by pressing **RTN**), simply position the cell display to the appropriate column or row, press **/** **H** **C** or **/** **H** **R**, then press **EDIT**. The entire header will be displayed without brackets. You can now use the HP-75 editing keys to make the change, or you can press **CLR** and type in the new header.

To return to the default headers, position the cell display to the correct column or row, press **/** **H** **C** or **/** **H** **R**, and press **CLR** **RTN**.

## Creating and Editing Cell Entries

VisiCalc provides several options for you to use when creating and editing your worksheet entries. If you make an error as you are typing entries into the worksheet, the HP-75 editing keys can be used to correct your entry. Or, if you decide to change the contents of a particular cell (for “what if” assumptions), Edit level makes it easy for you to modify complicated expressions right in place without “rewriting” the entire cell contents.

Edit Level Operations

As you are typing entries into VisiCalc, the HP-75's editing keys enable you to correct any mistakes. Once a label or value entry has been started, the cursor appears in the display, and the following editing keys are active.

Key	Operation
ATTN	Discards the current cell entry, restores the previous entry, and returns to Top level.
SHIFT ATTN	Exits VisiCalc.
TIME	Expands the view window by exchanging the user-defined headers in the cell coordinates field with the corresponding default headers. Pressing TIME again restores the user-defined headers. This mode is maintained with the worksheet.
→	Moves the edit cursor one character position to the right.
←	Moves the edit cursor one character position to the left.
SHIFT →	Moves the edit cursor past the last character position of the current entry.
SHIFT ←	Moves the edit cursor to the first character position of the current entry.
CTL →	Right-shifts the edit cursor one window width or to the end of the cell contents field, whichever distance is shorter.
CTL ←	Left-shifts the edit cursor one window width or to the beginning of the cell contents field, whichever distance is shorter.
I/R	Exchanges the replace cursor (■) with the insert cursor (⚡). Characters that are entered after this keystroke will be inserted to the left of the cursor.
SHIFT I/R	Causes the next keystroke or keystroke combination to display the character associated with it, regardless of key redefinitions.
DEL	Deletes the character under the cursor and moves the remaining characters to the left.
SHIFT DEL	Deletes the character under the cursor and all characters to the right of the cursor.
CLR	Clears the cell contents field.
!	Evaluates a formula, during value entry, before it is stored in a cell. The formula will be over-written by the result and cannot be recovered if RTN is pressed. (Described in section 3, page 79.)
BACK	Replaces the character to the left of the edit cursor with a blank, then moves the cursor one position to the left. In insert mode, the characters to the right of the cursor are moved one position to the left.
RTN	Stores the cell entry and returns to Top level.

### Editing Previously Entered Cell Contents

If you wish to edit the cell contents field after a cell entry has been stored (that is, the **RTN** key has been pressed), you can do so by pressing **EDIT**. The cursor then appears in the display, and the same editing keys are active as during cell entry. **EDIT** places the cursor on the first position of the cell contents field. If the cell contains a formula, the formula is displayed. If the cell contains a value, the value is displayed unformatted.

**Example:** Enter the following data into the **COSTS** worksheet started above. The VisiCalc display should be at the cell defined as **Qtr 1^Labor**.

COSTS	Qtr 1	Qtr 2
Labor	48400	49600
Material	56700	58900
# Units	11000	14500
Unit Cost	?	?

The unit cost in each quarter is found by adding the labor and material cost of that quarter and dividing by that quarter's number of units.

#### Input/Result

48400 **RTN**

Enters the first quarter's labor costs.

Qtr 1^Labor: 48400

↓ 57600 **RTN**

Oops! The wrong value was entered (57600 instead of 56700).

Qtr 1^Material: 57600

**EDIT** → 67 **RTN**

Enters Edit mode, moves the cursor to the second digit, and corrects the value.

Qtr 1^Material: 56700

↓ 11000 **RTN**

Enters the first quarter's number of units.

Qtr 1^# Units: 11000

⬇ ([Qtr 1^labor]+[Qtr 1^  
material])/[Qtr 1^# units]  
⬅ ⬅ ⬅ [I/R] i [RTN]  
Qtr 1^Unit Cos: 9.55454545455

Another error! This time units is misspelled.

Corrects a mistake while entering a formula. Remember that only 8 characters of the header are displayed.

If you are entering a formula using user-defined headers that are longer than 8 characters, you must enter only the first 8 characters of the header. If more than 8 characters are entered or the entry is otherwise incorrect, the HP-75 will beep when [RTN] is pressed, and place the cursor on the first place the error is detected.

**Input/Result**

[SHIFT] ⬆  
➡ 49600 [RTN]  
Qtr 2^Labor: 49600

Goes to the top of the worksheet.

Goes to cell Qtr 2^Labor and enters the labor cost.

⬇ 58900 [RTN]  
Qtr 2^Material: 58900

Enters the material cost in quarter 2.

⬇ 14500 [RTN]  
Qtr 2^# Units: 14500

Enters the number of units in quarter 2.

⬇ ([Qtr 2^labor]+[Qtr 2^  
material])/[Qtr 2^# units] [RTN]  
Qtr 2^Unit Cos(V): Qtr 2^# unit

Tries to enter the unit cost formula for quarter 2. The HP-75 will beep and place the cursor on the first character where an error was detected.

[SHIFT] ➡ ] [RTN]  
Qtr 2^Unit Cos: 7.48275862069

Corrects the unit cost formula and enters it into the cell.

## Repeating Labels

You can cause a character or group of characters to be repeated to form a label within a cell by using the Repeating Label command, /-.

When you press  $\boxed{/}$  followed by  $\boxed{-}$ , VisiCalc responds by placing  $\boxed{/-}$  after the cell coordinate field. Any entry you then type is treated as a label, and when the cell is displayed on a video monitor or printed, the entire cell is filled with repetitions of the label you typed in, regardless of the width of the cell. The HP-75's display shows only the entry that you typed in. As with other entries, repeating label entry is terminated with  $\boxed{RTN}$ , and all of the normal editing keys are active.

For example, typing  $\boxed{/}$   $\boxed{-}$   $\boxed{=}$  followed by  $\boxed{RTN}$  puts = in the cell along with a command to repeat the label when the cell is printed or displayed on a video monitor. When printed, the cell will display =====, etc. Following  $\boxed{/}$   $\boxed{-}$  with a dash,  $\boxed{-}$ , will fill the displayed cell with dashes, which results in horizontal lines in the worksheet.

## Using Redefined Keys

The HP-75 DEF KEY command (described in section 10, "Redefining the Keyboard" in the *HP-75 Owner's Manual*) can be used with VisiCalc to redefine keys for easier value entry. Suppose that cell Qtr 2^Labor is used several times in a worksheet. If the  $\boxed{CTL}\boxed{Z}$  key is defined as the character string Qtr 2^Labor, then each time Qtr 2^Labor is needed in a formula or a cell reference, you would key in a left bracket, press  $\boxed{CTL}\boxed{Z}$  (instead of typing the cell coordinates), then key in a right bracket.

The key redefinition must occur outside of the VisiCalc program. The redefined keys are active with VisiCalc whenever the edit cursor is blinking in the display.

A redefined key cannot be the first entry in the cell contents field, since the edit cursor is not in the display until you begin value entry. We could have defined  $\boxed{CTL}\boxed{Z}$  as  $\boxed{[Qtr\ 2^Labor]}$  (notice that brackets have been added). However, we would have had to begin value entry with some valid value entry key or character before using  $\boxed{CTL}\boxed{Z}$ .

## Modifying the Cell Display

Since your window into the worksheet is the single line display window of the HP-75, VisiCalc provides several ways for you to modify or change the information shown by the cell display to fit your particular requirements. These capabilities take the form of several *display modes*, each of which changes some aspect of the information in the cell display. When one of these display modes is active, it affects every cell in the worksheet equally. In addition, the modes are all stored with the worksheet file.



## Displaying a Formula or a Result

In section 1 (page 21) we introduced the concepts of Formula and Result Display modes. The **APPT** key is used to toggle between these two modes. Newly created worksheets are set to Result Display mode.

In Formula Display mode, a complex formula containing several user-defined cell references may be too long to be completely viewed within the confines of the contents field of the cell display. You could use the **RUN**, **SHIFT** **RUN**, **TAB**, and **SHIFT** **TAB** keystrokes to move the cell contents into the contents field for viewing, but another possibility exists. The **TIME** key and **SHIFT** **TIME** keystrokes shorten the cell display by converting user-defined headers in the display into their corresponding default headers.

## Changing Coordinate Display Modes

When VisiCalc is in Formula or Result Display mode, the **TIME** key toggles the *cell coordinates field* of the cell display between the user-defined headers and the corresponding default headers.

In Formula Display mode, the **SHIFT** **TIME** keystroke toggles the coordinates in the *cell contents field* of the cell display between the user-defined headers and the corresponding default headers. For example, to view the formula in cell Qtr 2^Unit Cost do the following:

### Input/Result

**APPT**

Sets Formula Display mode.

Qtr 2^Unit Cos: ([Qtr 2^Labor]+C

**SHIFT** **TIME**

Changes the headers in the cell contents field to the corresponding default headers (sets Default Coordinate Display mode).

Qtr 2^Unit Cos: (B1+B2)/B3

To go back to the user-defined headers (User-Defined Coordinate Display mode):

### Input/Result

**SHIFT** **TIME**

Changes the values in the cell contents field back to their user-defined headers. Notice that VisiCalc is still in Formula Display mode.

Qtr 2^Unit Cos: ([Qtr 2^Labor]+C

Suppose that a formula, which requires 20 characters, is entered using default headers. When VisiCalc converts the cell references in the formula to user-defined headers, it requires 100 characters to display the formula after **[RTN]** is pressed. Since a maximum of 91 characters can be displayed in a formula, VisiCalc will not be able to display the entire formula with user-defined headers—although the worksheet will recalculate correctly and VisiCalc will display the first 91 characters of the formula.

If you try to edit the formula while still in User-Defined Coordinate Display mode, VisiCalc will not accept the edited entry when you press **[RTN]** if the formula still contains more than 91 characters. The HP-75 will beep and place the cursor on the 92nd character, and that 92nd character will be a ?.

To store the edited formula in the cell, do either of the following:

- Reduce the number of characters in the formula to 91 or less, and press **[RTN]** to store the formula.
- Press **[ATTN]** to go to Top level, then press **[SHIFT][TIME]** to display the default headers. Edit the formula using the default headers, and press **[RTN]** to store the formula.

## The Display Format Toggle

When a worksheet is first created, VisiCalc displays all results to the full 12-digit precision of the HP-75. When you are doing calculations in cells, however, you often don't need (or want) to see the full precision of the result. VisiCalc provides commands that allow you change the format of the displayed result—to set Formatted Display mode. There are four commands that change the number of digits in the cell display to either Integer or Dollars format. (We will defer a complete discussion of these commands until section 8, "Formatting the Worksheet," page 131.) For now, you can change the display format by doing one of the following:

- Press **[/][F][I]** (*Local Integer format*) to round only the value in the current cell to an integer.
- Press **[/][F][\$]** (*Local Dollars format*) to round only the value in the current cell to two decimal places.
- Press **[/][G][F][I]** (*Global Integer format*) to round the values in all cells in the worksheet (except those that have a format set with the **/F** command) to an integer.
- Press **[/][G][F][\$]** (*Global Dollars format*) to round the values in all cells in the worksheet (except those that have a format set with the **/F** command) to two decimal places.

After setting the display to Integer or Dollars format, you can press **[SHIFT][APPT]** to switch the display to Unformatted Display mode. Pressing **[SHIFT][APPT]** a second time toggles the cell display back to the previous Formatted Display mode. The display format setting of a worksheet is stored with the worksheet.

**Example:** In the COSTS worksheet, change the Qtr 2^Unit Cost cell to display dollars and cents.

**Input/Result**

APPT

Qtr 2^Unit Cos17.48275862069

/ F \$

Qtr 2^Unit Cos1 7.48

Sets Result Display mode.  
  
Sets Dollars format in the  
Qtr 2^Unit Cost cell.

To display the unformatted value in the Qtr 2^Unit Cost cell:

**Input/Result**

SHIFT APPT

Qtr 2^Unit Cos1 7.48275862069

Sets Unformatted Display mode.

## Column or Row Manipulations

VisiCalc has three commands that enable you to change a worksheet by inserting, deleting, or moving any column or row.

### Inserting a Column or Row

The /I (*Insert*) command makes it easy for you to insert a column or row into the worksheet. To do so:

1. Move the cell display to the column that is to be just to the right of the new column, or the row that is to be just below the new row.
2. Press / I. VisiCalc prompts:

Insert: Col Row

3. Do either of the following:
  - Press C to insert a column just to the left of the column where the cell display is currently located.
  - Press R to insert a row just above the row where the cell display is currently located.

After you press **[C]** or **[R]**, the `Working` message is displayed while inserting is accomplished; afterwards VisiCalc displays the coordinates of the column or row just inserted.

When inserting, you can place the cell display anywhere in the column or row, just as long as the cell display is somewhere in the column that is to follow the inserted column, or in the row that is to be beneath the inserted row.

Inserting a column or row moves subsequent columns or rows (including any user-defined headers) to the right or down. Any cell references are automatically rewritten to reflect the new column/row status.

VisiCalc cannot insert a column or row if the active boundaries of the worksheet include column 255 or row 255. If you try to insert a 256th column (or row), VisiCalc will beep and remain at the `Insert: Col Row` prompt.

**Example:** In the `COSTS` worksheet, insert a row just above the unit cost row called `Tot Cost`.

Input/Result

**[ / ] [ I ]**

Executes the Insert command.

Insert: Col Row

Displays the insert options.

**[ R ]**

Selects the row option.

Qtr 2:4:

Inserts a new row 4.

**[ / ] [ H ] [ R ]**

Selects the row header option.

Row header 4:

Tot Cost **[ RTN ]**

Enters the row header `Tot Cost`.

Row header 4: [Tot Cost]

**[ ATTN ]**

Returns to Top level.

Qtr 2^Tot Cost:

### Deleting a Column or Row

You can use the **/D** (*Delete*) command to delete a column or row from the worksheet. To do so:

1. Move the cell display to any place along the column or row to be deleted.
2. Press **/** **D**. VisiCalc prompts:

Delete: Col Row

3. Do either of the following:
  - Press **C** to delete the *column* on which the cell display lies.
  - Press **R** to delete the *row* on which the cell display lies.
4. After you press **C** or **R**, you must press **Y** to delete the column or row.

All cell references affected by the deleted column or row are automatically rewritten to reflect the new column/row status. All subsequent columns or rows are moved to the left or up. The user-defined headers for the deleted column or row are also deleted.

**Example:** In the COSTS worksheet, delete the Ytd column.

#### Input/Result

<b>SHIFT</b> <b>&gt;</b> [Ytd] <b>RTN</b>	Goes to column Ytd.
<div>Ytd^Tot Cost:</div>	
<b>/</b> <b>D</b>	Executes the Delete command.
<div>Delete: Col Row</div>	Displays the delete options.
<b>C</b>	Selects the delete column option.
<div>Delete col [Ytd]: Yes No</div>	Asks if you really want to delete the column.
<b>Y</b>	Deletes the column.
<div>CC^Tot Cost:</div>	Ytd is deleted and subsequent columns are renamed and moved one position to the left.

Deleting a column or row that contains cells referenced by other cells will generate `ERROR` in the cell contents field of the dependent cells. When you look at the formulas whose values have become `ERROR`, the references to deleted cells will be underlined default coordinates. For example, suppose the formula in a cell is

```
[Jan^Income]+[Feb^Income]+SUM([Jan^House]...[Mar^House]),
```

and the Jan column is deleted. The formula would become

```
A1+[Feb^Income]+SUM(A3...[Mar^House]).
```

To delete the underlined cell references, either press `[EDIT][RTN]` or re-type the underlined cell reference and press `[RTN]`.

## Moving a Column or Row

The `/M` (*Move*) command enables you to move any column or row on the VisiCalc worksheet to any other column or row location. To do so:

1. Move the cell display to any cell in the column or row to be moved, and press `[/] [M]`. VisiCalc prompts:

```
Move: Col Row
```

2. Do one of the following:

- Press `[C]` to move a column.
- Press `[R]` to move a row.

3. VisiCalc then prompts for the destination (the “to” column or row coordinates). The destination column or row coordinates can be user-defined or default.

When moving a column (or row), all columns (or rows) between the source and the destination, including the destination, will move one position toward the source, and the source will be placed in the destination’s previous location.

All cell references affected by the moved column or row are rewritten to reflect the new column/row status.\* The column or row user-defined headers are also moved.

When using the Move command, if you specify an improper coordinate (such as a row coordinate after `[/] [M] [C]` has been pressed), the HP-75 beeps, moves the cursor to the first character of the cell contents field, and waits for an appropriate response.

---

\* After the move, cell references point to the new locations of the old data. This may lead to forward references (refer to “Forward and Circular References” in section 3).

**Example:** In the COSTS worksheet, move the # Units row beneath the Tot Cost row.

**Input/Result**

SHIFT > [ # units ] RTN

CC# Units]:

Goes to row # Units.

/ M

Move: Col Row

Executes the Move command.

R

Move row [ # Units ] to: █

Displays the move options.

[ tot cost ] RTN

CCTot Cost]:

Selects the row move option.

Prompts for the row destination.

Defines the new row destination.

Displays the current cell after the row is moved.

The printed worksheet now looks like this (the header and column widths are 10):

COSTS	A	B
	Qtr 1	Qtr 2
1 Labor	48400	49600
2 Material	56700	58900
3 Tot Cost		
4 # Units	11000	14500
5 Unit Cost	9.5545455	7.48



## Section 3

# Using Functions and Formulas

## Contents

What You'll Learn in This Section .....	69
Expressions and Numbers .....	70
Mathematical Hierarchy .....	71
Scientific Notation .....	71
Precision of Calculations .....	72
VisiCalc Functions .....	72
Parameter Lists and Ranges .....	73
The SUM Function .....	73
The MAXL and MINL Functions .....	74
The AVERAGE or MEAN Function .....	76
Propagating NA and ERROR .....	77
HP-75 Functions and VisiCalc .....	78
Precalculation of an Expression .....	79
Recalculating the Worksheet .....	79
Recalculation Commands .....	80
Forward and Circular References .....	83

## What You'll Learn in This Section

In this section you will learn how to use VisiCalc functions and formulas. When you finish this section, you should be able to:

- Create arithmetic expressions using operators, numbers, cell references, and functions.
- Know what scientific notation is and when it is used.
- Understand the mathematical hierarchy of VisiCalc and HP-75 mathematical operations.
- Understand and specify *parameter lists* and *ranges* in function calls.
- Use the SUM, MAXL, MINL, MEAN, and AVERAGE functions in expressions.
- Understand the operation and use of the NA and ERROR functions.
- Precalculate an expression.
- Turn Automatic Recalculation mode on and off.
- Change the order in which recalculation occurs.

- Manually recalculate the entire worksheet at any time.
- Understand *forward* and *circular references* and how to avoid them.

Before reading this section, you should have read and understood the information in sections 1 and 2.

## Expressions and Numbers

Generally, an expression in VisiCalc consists of a series of one or more *operands* separated by zero or more arithmetic *operators*. An operand can be any of the following:

- A number, which can have a decimal point and/or an exponent.
- A cell reference, obtained by typing the cell coordinate.
- A function.
- Any of the above combined by one or more operators (arithmetic, relational, or logical).

Operators can be any of the following:

- Arithmetic Operators

+	Addition.
-	Subtraction.
*	Multiplication.
/	Division.
DIV or <span>CTL</span> <span>/</span>	Integer division.
^	Exponentiation.

- Relational Operators

=	Equal to.
<	Less than.
<=	Less than or equal to.
>	Greater than.
>=	Greater than or equal to.
<> or #	Not equal to.

- Logical Operators

A AND B	Evaluates to 1 (true) if and only if both A and B are nonzero (true).
A OR B	Evaluates to 1 (true) if A or B or both are nonzero (true).
A EXOR B	Evaluates to 1 (true) if A is nonzero (true) or B is nonzero (true) but not both.
NOT A	Evaluates to 1 (true) if A is 0 (false).

**Note:** When used with default coordinates, the logical operators require that spaces be included as above.

## Mathematical Hierarchy

The mathematical hierarchy of VisiCalc is the same as that in the HP-75. The table below lists the HP-75 operators in their order of precedence. Expressions are evaluated from left to right for operators at the same level.

### Performed first.

( )	Nested parentheses are evaluated from the inside out.
functions	
^	
NOT	
*, /, DIV or \	
+, -	
=, >, >=, <, <=, <> or #	
AND	
OR, EXOR	

### Performed last.

If you are in doubt about how an expression will be evaluated by VisiCalc, add parentheses to clarify your understanding. If you add unneeded parentheses, they are automatically removed when the expression appears on the display. Parentheses may be nested, so even an expression such as  $700 + ((100 - (35 * 45)) / (11 * 12))$  can be easily evaluated. In VisiCalc, parentheses can be nested 30 pairs deep. However, a value entered can only have a maximum of 91 characters.

For further information on mathematical hierarchy, refer to section 5 of the *HP-75 Owner's Manual*.

## Scientific Notation

Scientific notation is a shorthand system for expressing numbers that are too large or too small to fit the display normally—that is, numbers that cannot be expressed adequately with 12 digits. When a number is displayed in the *general format* (the format used when the worksheet is first created), VisiCalc will automatically shift between conventional and scientific notation as required to display the value with the greatest precision.

**Example:** The number 1 followed by 20 zeros would be expressed as: 1E20.

For more information on scientific notation, refer to “Exponential Notation” in section 4 of the *HP-75 Owner’s Manual*.

## Precision of Calculations

Newly created worksheets display all numbers to the full precision of the HP-75 (12 digits). Although VisiCalc allows you to *display* numbers to other degrees of precision,\* no matter how you alter the appearance of the worksheet numbers, VisiCalc always computes with numbers that are maintained to the full internal precision of the HP-75.

You may wish to use a *rounding formula* to reduce the actual stored precision of a calculated value. For example, to reduce the precision of a value to just two decimal places (for dollars and cents calculations), the formula is:

$$IP(COORDINATE*100+.5)/100$$

where *coordinate* is the cell coordinate of the value to be rounded.

For more information on the precision of calculation, refer to “Numeric Precision” in section 4 of the *HP-75 Owner’s Manual*.

## VisiCalc® Functions

As you know, you can use functions as part of the formulas you write into the VisiCalc worksheet. In fact you can use almost the entire set of available HP-75 numeric functions in your formulas, and you can also use functions that are unique to VisiCalc.

You will find a complete list of available HP-75 and VisiCalc functions in appendix B, “VisiCalc Functions” (page 175).

Remember that if the first character of a cell contents field is a letter— such as the S in SIN—it will be seen as a *label* by VisiCalc. To key in the *function* SIN, you must precede it by a mathematical symbol, such as +. An alternative is to precede the function with the “at” sign, @. The + and @ tell VisiCalc that the following entry is a value. The symbol appears in the cell contents field when you type it in, but it disappears when you press RTN to store the entry. The symbol is not shown when the formula is displayed again.

---

\* Integer format and Dollars format were mentioned in section 2 (page 63). In section 8, “Formatting the Worksheet” (page 131), we will give a complete discussion of how to change the format of the displayed result.

## Parameter Lists and Ranges

Most VisiCalc functions are designed to accept more than one value. The function call consists of the function name followed by a parameter list enclosed in parentheses. Parameter lists specify the values to be passed to a function.

A parameter specifies the cell(s) used by the function: it can be a single cell, a number, an expression, or a range (see below). A parameter list is a series of parameters separated by commas, and it can contain up to 255 parameters.

A range is any number of entries in adjacent columns (such as B3, C3, and D3) or adjacent rows (such as A2, A3, and A4). The entire range is a single parameter and is specified by keying in the coordinates of the first entry, followed by an ellipsis (...), then the coordinates of the last entry. For example, B3...D3 represents the entries B3, C3, and D3.

Examples of equivalent parameter lists are shown below.

A1, A2, A3, A4, A5	Cells A1, A2, A3, A4, and A5 (five parameters).
A1...A4, A5	Range A1 to A4 and cell A5 (two parameters).
A1, A2...A4, A5	Cell A1, range A2 to A4, and cell A5 (three parameters).
A1, A2...A5	Cell A1 and range A2 to A5 (two parameters).
A1...A5	Range A1 to A5 (one parameter).

If a range or parameter list contains a blank cell or a label, the cell is evaluated as zero.

## The SUM Function

The VisiCalc function **SUM** computes the sum of the values in all the cells specified in its parameter list.

**Example:** In the **BUDGET** worksheet, what is the total food allowance for the first 6 months of the year?

Reload the original **BUDGET** worksheet and use the **Go To** command to move to cell **G6Food1**. Then use the **/H** command to create a new column header **Total**. Since the **BUDGET** worksheet was loaded and altered in section 1, you will need to purge or rename it before reloading **BUDGET** from the card. Refer to the *HP-75 Owner's Manual* for these procedures.

**Input/Result**

[SHIFT] [>] g[food] [RTN]

Goes to cell G[Food].

G[Food]:

[/] [H] [C] Total [RTN]

Enters column header Total.

Col header G: [Total]

To compute the sum, you could laboriously type in [Jan^food]+[Feb^food]+[Mar^food]+[Apr^food]+[May^food]+[Jun^food]. But a faster way is to use the SUM function. Just type in SUM (preceded by a + or @ sign\*), followed by the entry range [Jan^food]...[Jun^food] in parentheses.

**Input/Result**

[ATTN]

Returns to Top level.

Total^Food:

+sum([Jan^food]...[Jun^food])

Enters the function and the parameter list for Total^Food.

[RTN]

Total^Food: 1045.00

Displays total food cost for the first six months—the result of the formula you just entered.

**The MAXL and MINL Functions**

The MAXL (*maximum in list*) function finds the maximum value in a parameter list, while the minimum value is found with the MINL (*minimum in list*) function.

**Example:** Find the minimum food bill for the months January through June.

\* Remember from section 1 that the @ sign can also be inserted to change a label entry to a value entry.

Input/Result



Goes to cell HCFoodJ.

HCFoodJ:



Least



Enters column header Least.

Col header H: [Least]



Returns to Top level.

Least^Food:

+min1([Jan^food]...[Jun^food])

Enters the minimum value function.



Least^Food: 140.00

Displays the least amount spent on food during the first six months.

BUDGET:	Jan	Feb	Mar	Apr	May	Jun	Total	Least
Income:	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00		
House :	400.00	400.00	400.00	400.00	400.00	400.00		
Insur :	100.00	100.00	100.00	100.00	100.00	100.00		
Taxes :	600.00	600.00	600.00	600.00	600.00	600.00		
Food :	200.00	175.00	180.00	155.00	195.00	140.00	1045.00	140.00
Util :	110.00	95.00	87.00	81.00	68.00	55.00		
TotExp:	1410.00	1370.00	1367.00	1336.00	1363.00	1295.00		
Remain:	590.00	630.00	633.00	664.00	637.00	705.00		



The AVERAGE or MEAN Function

AVERAGE and MEAN are two names for the same function. This function computes the *average* (or *arithmetic mean*) of the values in a parameter list. The average is the sum of the parameters divided by the number of numeric parameters. If the parameter list contains only labels or empty cells, AVERAGE and MEAN will both return ERROR.

**Example:** To find the average monthly utility expense in the BUDGET worksheet:

Input/Result



Goes to cell ICUtil1J.

ICUtil1J:

☒ ☐ ☐ Average ☐ RTN

Enters the new column header Average.

Col header I: [Average]

☐ ☒ +average([Jan^util],...  
[Jun^util]) ☐ RTN

Enters the average function.

Average^Util: 82.67

Displays the average utility bill for the first six months.

BUDGET:	Jan	Feb	Mar	Apr	May	Jun	Total	Least	Average
Income:	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00			
House :	400.00	400.00	400.00	400.00	400.00	400.00			
Insur :	100.00	100.00	100.00	100.00	100.00	100.00			
Taxes :	600.00	600.00	600.00	600.00	600.00	600.00			
Food :	200.00	175.00	180.00	155.00	195.00	140.00	1045.00	140.00	
Util :	110.00	95.00	87.00	81.00	68.00	55.00			82.67
TotExp:	1410.00	1370.00	1367.00	1336.00	1363.00	1295.00			
Remain:	590.00	630.00	633.00	664.00	637.00	705.00			

## Propagating NA and ERROR

If you enter a formula that, when evaluated, causes an arithmetic error to occur,\* the affected cell will display **ERROR**. In addition, this error will be *propagated* throughout the worksheet—that is, any formula or value reference that depends on the contents of a cell displaying **ERROR** will display the word **ERROR** when viewed. If you type **+ERROR** into a cell, the word **ERROR** will propagate to all other cells that reference that cell.

If an extension function (which will be described in section 10) doesn't return a value, **ERROR** will be placed in the current cell.

Similarly, if you do not know the value needed for a particular cell, you can simply type in **+NA** for *not available*. Like **ERROR**, the **NA** display is propagated over the entire worksheet—wherever there is a value or formula that depends on the contents of the cell.†

**Example:** In the **BUDGET** program, change the value for June's income to **NA** and see what happens to the worksheet:

### Input/Result

[SHIFT][>] [Jun^income] [RTN]

Goes to cell Jun^Income.

Jun^Income: 2000.00

+na [RTN]

Enters NA in cell Jun^Income.

Jun^Income: NA

[SHIFT][↓]

Goes to the bottom of the worksheet (cell Jun^Remain).

Jun^Remain: NA

\* Arithmetic errors are listed in appendix E of the *HP-75 Owner's Manual*.

† If both **ERROR** and **NA** occur in a formula, the cell will display the symbol that occurs first.

The NA propagates to every cell containing a formula or value reference that depends on the contents of cell Jun^Income (cell F1). The worksheet now looks like this:

BUDGET:	Jan	Feb	Mar	Apr	May	Jun	Total	Least	Average
Income:	2000.00	2000.00	2000.00	2000.00	2000.00	NA			
House :	400.00	400.00	400.00	400.00	400.00	NA			
Insur :	100.00	100.00	100.00	100.00	100.00	100.00			
Taxes :	600.00	600.00	600.00	600.00	600.00	NA			
Food :	200.00	175.00	180.00	155.00	195.00	140.00	1045.00	140.00	
Util :	110.00	95.00	87.00	81.00	68.00	55.00			82.67
TotExp:	1410.00	1370.00	1367.00	1336.00	1363.00	NA			
Remain:	590.00	630.00	633.00	664.00	637.00	NA			

### HP-75 Functions and VisiCalc®

In addition to the functions that are found only in VisiCalc, you can also use nearly the entire set of mathematical functions provided in your HP-75. Any numeric function that takes only numeric parameters and returns a single numeric result can be used in a VisiCalc cell entry. Like the VisiCalc functions, if an HP-75 function is the first item in a cell entry, you must precede it with a + or @ sign, so that VisiCalc knows you are entering a value rather than a label. Refer to appendix B of this manual for a complete list of HP-75 functions usable with VisiCalc.

You can also use the entire set of HP-75 logical and relational operators, including =, >, <, >=, <=, <> or #, AND, OR, EXOR, and NOT. This means that powerful expressions or formulas are possible without requiring special functions.

**Example:** Suppose you are creating a worksheet that shows customer order information for a product, and you want the discount on the purchase price to be 7% if the customer orders 10,000 units or less, and 12% if more than 10,000 units are ordered. You can create a cell entry that gives the purchase discount—and reflects this pricing structure—by writing a conditional expression like the one below (assume that the order quantity is in cell A1).

```
A1*(.07*(A1<=10000)+.12*(A1>10000))
```

The relational operators, `<=` and `>`, return 0 if the expression is false or 1 if the expression is true. Since only one of the expressions in the formula above can be true, only one of the multiplications inside the outer parentheses will produce a nonzero result. For example, if a customer orders 5,000 units (A1 equals 5,000) the formula above will become:

$$5000*(.07*(1)+.12*(0))$$

This further reduces to  $5000*.07$ —the correct discount for the order quantity!

## Precalculation of an Expression

Sometimes you may need to do a calculation to get a result that you want to place in a cell entry. If you aren't interested in storing the formula used in the calculation—only the result—you can use the `[!]` to *precalculate* the formula. When you press `[!]` after keying in a formula but before pressing `[RTN]`, the formula will be evaluated *before* it is entered into the cell. Thus, when you are entering formulas, `[!]` lets you use the display like a simple calculator.

For example, add the electricity bill (\$78) to the phone bill (\$26) and the water bill (\$15). Then enter the result in cell Jan^Util of the BUDGET worksheet.

### Input/Result

`[SHIFT][>][Jan^util][RTN]`

Goes to cell [Jan^Util].

Jan^Util: 110.00

78+26+15 `[!]`

Pressing `[!]` calculates the expression and returns the result to the display.

Jan^Util(V): 119■

`[RTN]`

Enters the value into cell A7.

Jan^Util: 119.00

You can see that only the number is in the cell—the colon is not underlined.

## Recalculating the Worksheet

As your VisiCalc worksheets and calculations grow larger and more complex, there may be significant delays as you enter values or formulas into cells. This is because every time you press `[RTN]`, the entire worksheet is recalculated, and any cells that reference the current cell are reevaluated. Each function such as `LOG` and `SIN`, for example, takes a fraction of a second to evaluate, and a call to an extension function can take several seconds to recalculate. The delay increases as the size and complexity of the worksheet increases.

## Recalculation Commands

In many cases, you don't need to have all the values recalculated every time you change an entry. The **/GR** (*Global Recalculation*) command enables you to change several entries and then recalculate when you're ready to look at the results. To use the Global Recalculation command:

1. Press **/ G R**. VisiCalc prompts:

Recalc: Auto Manual

2. Do either of the following:

- Press **A** to set Automatic Recalculation mode and to recalculate the entire worksheet.
- Press **M** to set Manual Recalculation mode.

When a worksheet is first created, Automatic Recalculation mode is set. The recalculation mode field in the status line (page 31) contains the recalculation mode indicator: it can be either **A**, for automatic recalculation, or **M**, for manual recalculation.

Since Automatic Recalculation mode causes the entire worksheet to be recalculated whenever a new value is entered, manual recalculation allows faster data entry than automatic recalculation. The entire worksheet is recalculated whenever you type an exclamation point, **!**, while VisiCalc is at Top level.

VisiCalc evaluates the formulas on the worksheet in one of two possible orders: column-by-column or row-by-row. A newly created worksheet is set to recalculate "columns first." It will evaluate **A1** first, then **A2**, **A3**, etc., then **B1**, **B2**, **B3**, etc., then **C1**, and so on. If you change the recalculation order to "rows first," (as described below) VisiCalc will evaluate **A1** first, then **B1**, **C1**, etc., then **A2**, **B2**, **C2**, etc., then **A3**, and so on. The recalculation order field in the status line (page 31) contains the recalculation order indicator; it can be either **C**, for recalculation column-by-column, or **R**, for recalculation row-by-row.

The **/GO** (*Global Order*) command specifies the order of the recalculation—column-by-column or row-by-row. To use the Global Order command:

1. Press **/ G O**. VisiCalc prompts:

Order of recalc: Col Row

2. Do either of the following:

- Press **C** to set the recalculation order to be column-by-column, beginning with column A. The entire worksheet is recalculated if Automatic Recalculation mode is set.
- Press **R** to set the recalculation order to be row-by-row, beginning with row 1. The entire worksheet is recalculated if Automatic Recalculation mode is set.

For many problems, the choice of column-by-column or row-by-row recalculation has no effect on the results. But there are times when you must use the correct recalculation mode to obtain correct results, and it's important to recognize these cases when they arise.

Let's consider a simple example. Exit the current BUDGET worksheet, and type the following formulas into a new worksheet named TEST. As you are entering the formulas, think about how each entry depends on the others.

TEST	A	B
1	1	2•A2
2	2•A1	2•B1

Input/Result

VC

Enters VisiCalc.

Enter worksheet name: BUDGET

Prompts for a worksheet name.

test

Enters a new worksheet named TEST.

A1:

1

Enters a value into cell A1.

A1: 1

2\*A1

Enters a formula into cell A2.

A2: 2

2\*B2

Goes to cell B1 and enters a formula.

B1: 4

2\*B1

Goes to cell B2 and enters a formula.

B2: 8

In this example, the group of entries starting at A1 must be recalculated in column order: A1, A2, B1, B2 (since B1 depends on A2).

Now change the recalculation order from column-by-column to row-by-row by using the **/GOR** command, and specify manual recalculation with the **/GRM** command.

**Input/Result**

**/** **G** **O**

Prompts for the order of recalculation.

Order of recal: Col Row

**R**

Selects Row Recalculation mode.

B2: 8

**/** **G** **R**

Prompts for automatic or manual recalculation.

Recalc: Auto Manual

**M**

Selects Manual Recalculation mode.

B2: 8

Change the value of cell A1 to 2, perform the manual recalculation with a single press of **!**, and view the result in cell B2.

**Input/Result**

**↑** **←** 2 **RTN**

Goes to cell A1 and enters a new value.

A1: 2

**!**

Recalculates the worksheet.

A1: 2





B2: 8

Notice that the value in B2 is unchanged, even though A1 contains a new value. This happens because cell B1 is recalculated before cell A2 (and B2 equals  $2 \times B1$ ).

Now, do another manual recalculation.

### Input/Result




B2: 16

The correct value is now displayed because cell B1 has been updated.

This example shows that you should plan your calculations with either column-by-column or row-by-row recalculation in mind, but not both. If possible, you should arrange things so the results will be correct with either recalculation order.

In general, try to construct your worksheet so that all cells with value references will be recalculated only *after* the cells on which they depend. This is done by locating these “dependent” cells below and to the right of all their value references. Also, be careful when creating formulas referring to cells that, in turn, refer to other cells. Such a situation could lead to a *forward* or *circular reference*, discussed below.

If you find yourself with a problem of conflicting requirements for column-by-column and row-by-row recalculation, you can deal with these requirements by pressing  for an extra recalculation each time you change a value and look for recalculated results. But first make certain that your column and row conflict is not actually due to a forward reference or a circular reference.



## Forward and Circular References

VisiCalc is powerful and extremely simple to use, but there are a few problems that can occur if you are not careful about how you set up the worksheet. Two of these problems are *forward references* and *circular references*.

A cell with a forward reference is one whose value will change *after* the cell is recalculated (as in the previous example). The only way to achieve the correct results throughout the worksheet is to perform a second recalculation. In extreme cases, forward references may refer to other forward references, so that correct results can be obtained only after several recalculations.

An effect even more startling is caused by a circular reference. The value of such a formula cannot be determined with any number of recalculations!

For example, suppose that you are projecting future profits, taking into account various revenues and expenses. One of the expenses is employee salaries, and one part of salaries consists of profit-sharing. Unless you are careful, you may create a circular reference: salaries with profit-sharing depend on profits, but profits depend on expenses including salaries. (To resolve this circularity, you must calculate a figure for profits before profit-sharing is taken out.)

You can often discover forward or circular references because pressing  repeatedly does not give you a consistent result. You can thus use  to check for forward and circular references.

Once you are aware of the issues involved in recalculation, it is not difficult to avoid conflicts in the order of recalculation of forward and circular references. Generally these problems arise when you are working too quickly without a plan, or “patching” an existing layout on the worksheet. If your approach to the problem is orderly and well thought-out, it is unlikely that you will encounter any problems with recalculation.

## Using the Replicate Command

### Contents

What You'll Learn in This Section .....	85
The Replicate Command Procedure .....	85
Replicating Numbers and Labels .....	88
Replicating a Formula .....	91
Using the Relative Option .....	91
Using the Both Option .....	93
Other Types of Replication .....	96
Replicating a Range of Cells .....	96
Replicating Into a Range of Columns or Rows .....	98

### What You'll Learn in This Section

In this section you will learn how to use the **/R** (*Replicate*) command. The function of the Replicate command is to repeat an entry across a column or row, or to repeat an entire column or row on the VisiCalc worksheet. When you finish this section, you should be able to:

- Understand and use the *Replicate* command.
- Understand *source* and *target ranges*.
- Use *relative replication* and *replication without changes*.
- Replicate formulas.
- Replicate a cell into a range of cells.
- Replicate a column or row into another column or row.
- Replicate a column or row into a range of columns or rows.

Before reading this section, you should have read and understood the information in sections 1, 2, and 3.

### The Replicate Command Procedure

With the Replicate command, a single cell can be replicated either into another single cell, or into a column or row of adjacent cells. A single column or row can be replicated either into another column or row, or into a range of adjacent columns or rows.

The Replicate command is one of the most useful and powerful commands in VisiCalc. With replication, you can cause cell entries such as formulas, labels, values, or cell references to be copied into other columns or rows. (Cell formats can also be replicated. Formatting is discussed in section 8, “Formatting the Worksheet”.) The Replicate command saves you the time you would otherwise spend laboriously copying these entries into each cell.

Here is the replication procedure:

1. Position the cell display on the first cell of the source range (the first cell to be replicated) and type `/ [R]`. VisiCalc prompts:

`Repl source: coordinates of the current cell*`

2. Do one of the following:
  - If the current cell is the entire source range, press `[RTN]`.
  - If the current cell is the first cell in the source range, press the decimal point key `[.]` three times to generate a three-dot ellipsis, then type in the coordinates of the last cell in the source range.
  - If the current cell is not the first cell in the source range, press `[CLR]` to clear the cell contents field. Then type in the coordinates of the first cell of the source range before entering the ellipsis (...) and the final cell of the source range.
3. VisiCalc now prompts `Repl target: █` to ask you to enter the target coordinates (that is, the coordinates of the cell or cells into which the source range is to be replicated). As with the source range, the coordinates of the first and last cells of the target range must be separated by an ellipsis. Enter the target range and press `[RTN]`.

If you wish to view the source range while being prompted for the target range, you can do so by pressing the up-arrow key (`[↑]`). When the source range is in the display, you can use the `[TAB]`, `[SHIFT][TAB]`, `[RUN]`, `[SHIFT][RUN]`, and `[TIME]` keys to view the source range. To get back to the prompt for the target range, press the down-arrow key (`[↓]`). To exit the Replicate command and return to Top level, press `[ATTN]`.

4. If the cells in the source range to be replicated contain cell references, VisiCalc prompts:

`Repl: No change Relative Both`

This option will be discussed later under “Replicating a Formula,” (page 91).

When you are replicating a single cell into another single cell, the source and target ranges should both reference single cells. To replicate a single cell into each of several cells in a column or row of cells, the source range consists of the single cell and the target range should specify the first and last cells of the column or row. For example:

---

\* The type face used for *coordinates of the current cell* is employed throughout this manual for *parameters* that you can change or enter. In some instances VisiCalc will supply a default parameter, in others the parameter field will be blank until you enter a value. The use of this type face for parameters follows the convention established in the *HP-75 Owner's Manual*.

Replicating this source range	Into	This target range	Copies the contents of the cells as follows:
A1...A1		D4...D4	A1 into D4
A1...A1		B1...G1	A1 into B1, C1, D1, E1, F1, and G1
A1...A1		B1...B4	A1 into B1, B2, B3, and B4

Replication of a column into another column or a row into another row is done by indicating the first and last cells of the source range but just the *first* cell of each column or row of the target range. For example:

Replicating this source range	Into	This target range	Copies the contents of the cells as follows:
A1...A3		C1...C1	A1 into C1; A2 into C2; A3 into C3
A1...C1		A2...A2	A1 into A2; B1 into B2; C1 into C2

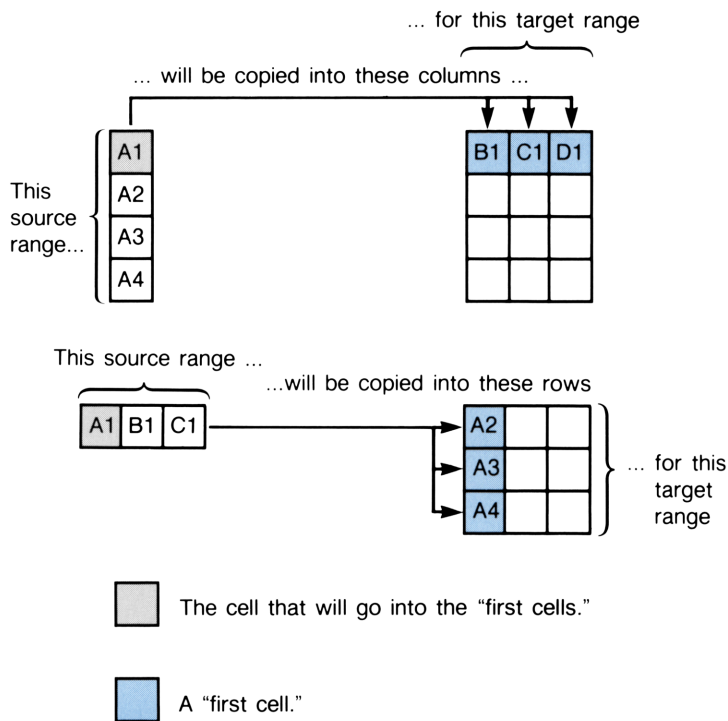
**Note:** Columns can only be replicated into one or more columns, and rows into one or more rows. A column cannot be replicated into a row, and a row cannot be replicated into a column.

VisiCalc can also replicate a column or row into a *range* of adjacent columns or rows. For this procedure the source range consists of the first and last cells of the column or row to be replicated. If a column is being replicated, the target range should consist of the first and last cells of the *row consisting of all the “first cells” of the target columns* (the cells that will receive a copy of the first entry of the source column). If a row is being replicated, the target range should consist of the first and last cells of the *column consisting of all the “first cells” of the target rows* (the cells that will receive a copy of the first entry of the source row).\*

The diagrams that follow (which show the first cells shaded) illustrate the following replications:

Replicating this source range	Into	This target range	Copies the contents of the cells as follows:
A1...A4		B1...D1	Column A1...A4 into columns B1...B4, C1...C4, and D1...D4
A1...C1		A2...A4	Row A1...C1 into rows A2...C2, A3...C3, and A4...C4

\* If the source range is a column, and the target range is also a column, the Replicate command will assume that you meant to specify only the first cell in the target range and will ignore the remainder of the target range. The same thing will happen if the ranges are rows.



If the cells specified in a source range or in a target range are not part of the same column or row, the HP-75 will beep and display the incorrect entry for correction. As with any other entry in the display, you can then correct or alter the Replicate command with the editing keys.

Replicating a column (or row) into another column (or row), when the source and target ranges overlap, may produce unexpected results.

## Replicating Numbers and Labels

**Example:** Let's expand the BUDGET worksheet to include months July through December (add columns JUL through DEC). Then, replicate the insurance row through the remaining months of the year.

First, reload the BUDGET worksheet from the card, except this time name the worksheet BUDGET1.

### Input/Result

SHIFT → →

Goes to column G.

GCIncomeJ:

/ H C

Executes the Column Header command.

Col header G:

Prompts for a new column header.

Jul RTN

Enters column header Jul.

Continue defining columns until all six have been created.

→ Aug RTN

→ Sep RTN

→ Oct RTN

Defines column headers Aug, Sep, Oct, Nov, and Dec.

→ Nov RTN

→ Dec RTN

The worksheet now looks like this:

BUDGET:	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Income:	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00						
House :	400.00	400.00	400.00	400.00	400.00	400.00						
Insur :	100.00	100.00	100.00	100.00	100.00	100.00						
Taxes :	600.00	600.00	600.00	600.00	600.00	600.00						
Food :	200.00	175.00	180.00	155.00	195.00	140.00						
Util :	110.00	95.00	87.00	81.00	68.00	55.00						
TotExp:	1410.00	1370.00	1367.00	1336.00	1363.00	1295.00						
Remain:	590.00	630.00	633.00	664.00	637.00	705.00						

Now, return to Top level and go to cell Jun^Insur.



Input/Result

ATTN

Returns to Top level.

Jul^Income:

SHIFT > [Jun^insur] RTN

Goes to cell Jun^Insur.

Jun^Insur: 100.00

You now have June’s insurance allowance in the display. In order to create a row that includes an insurance allowance for each month, you *could* move to cell Jul^Insur and type in 100, then move to cell Aug^Insur and do the same thing, and so forth through Dec^Insur. But it’s easier to use the Replicate command.

The entire source range for the replication is the single cell Jun^Insur. Since this cell only contains a number, the Repl: No change Relative Both prompt will not be displayed.

Input/Result

/ R

Executes the Replicate command.

Repl source: [Jun^Insur]

Prompts for the source range.

RTN

Defines the source range as Jun^Insur.

Repl target:

Prompts for the target range.

[Jul^insur]...[dec^insur] RTN

Defines the target range.

Jun^Insur: 100.00

Replicates the cell Jun^Insur and returns to the current cell.

Now, when you view the worksheet, you will see that June’s insurance cost has been replicated across the entire row.

Input/Result

→

Goes to cell Jul^Insur.

Jul^Insur: 100.00

SHIFT →

Goes to cell Dec^Insur.

Dec^Insur: 100.00

BUDGET1	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Income:	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00						
House:	400.00	400.00	400.00	400.00	400.00	400.00						
Insur:	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00
Taxes:	600.00	600.00	600.00	600.00	600.00	600.00						
Food:	200.00	175.00	180.00	155.00	195.00	140.00						
Util:	110.00	95.00	87.00	81.00	68.00	55.00						
TotExp:	1410.00	1370.00	1367.00	1336.00	1363.00	1295.00						
Remain:	590.00	630.00	633.00	664.00	637.00	705.00						

## Replicating a Formula

When you replicate a formula that contains a cell reference, VisiCalc prompts:

Repl: No change Relative Both

You must then decide whether you want the formula replicated exactly as written, with *no change*, whether you want the formula that is replicated into each succeeding cell to be changed so that it is *relative* to its new location in the worksheet, or whether you want the formula to have *both* options—some cell references relative and some simply copied.

## Using the Relative Option

In the BUDGET1 worksheet, let's replicate the formula in cell Jun^Income to cells Jul^Income through Dec^Income. If we simply replicate the formula with no change, the formula in cell Jun^Income (the value [May^Income]) would be repeated in each cell of the target range. But since we want the formula in Jul^Income to be relative to the value in Jun^Income (to refer to a value relative to its location, just like the formula in Jun^Income does), and Aug^Income relative to Jul^Income, etc., we'll specify *relative* replication.

### Input/Result

SHIFT > [Jun^Income] RTN

Goes to cell Jun^Income.

Jun^Income: 2000.00

[ / ] [ R ]

Executes the Replicate command.

Repl source: [Jun^Income]■

Prompts for the source range.

[ RTN ]

Defines the source range as the cell Jun^Income.

Repl target: ■

Prompts for the target range.

[Jul^income]...[Dec^income] [ RTN ]

Defines the target range.

Repl: No change Relative Both

Displays the replicate options, because a formula is being replicated.

Here, VisiCalc is prompting for the type of replication: **N** for *No change*, **R** for *Relative*, or **B** for *Both*. Since we want the replication to be relative, we proceed as follows:

**Input/Result**

[ R ]

Selects the relative option.

Jun^Income: 2000.00

[ → ]

Goes to cell Jul^Income.

Jul^Income: 2000.00

[ APPT ]

Displays the formula for cell Jul^Income.

Jul^Income: [Jun^Income]

[ SHIFT ] [ → ]

Goes to cell Dec^Income and displays the formula.

Dec^Income: [Nov^Income]

[ APPT ]

Displays the value for cell Dec^Income.

Dec^Income: 2000.00

BUDGET1	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Income:	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00
House :	400.00	400.00	400.00	400.00	400.00	400.00						
Insur :	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00
Taxes :	600.00	600.00	600.00	600.00	600.00	600.00						
Food :	200.00	175.00	180.00	155.00	195.00	140.00						
Util :	110.00	95.00	87.00	81.00	68.00	55.00						
TotExp:	1410.00	1370.00	1367.00	1336.00	1363.00	1295.00						
Remain:	590.00	630.00	633.00	664.00	637.00	705.00						

As you can see, the replication was a relative one, with the formula not merely *copied* in each cell, but actually depending on the value in a previous cell.

## Using the Both Option

When you replicate a formula that contains a cell reference and you select the Both option, VisiCalc will display the cell coordinates field, then a colon, and the formula in the cell. The first cell reference in the formula is underlined. The right-most portion of the display contains the prompt:

+N R el

You must respond with **N** (for no change) or **R** (for relative) for *each* cell reference in the entire source range. If a formula contains several cell references, you will see each one underlined individually as you respond with **N** or **R**. A formula that does not fit in 91 characters with user-defined headers will automatically be shown with default headers.

In the BUDGET1 worksheet, create a column **Total** that is the total income for the year.

### Input/Result



Goes to the column to be defined.

MCIncome]:



Executes the Header command and selects the column header option.

Col header M:

Total **RTN**

Enters header **Total**.

Col header M: [Total]

ATTN

Returns to Top level.

Total^Income:

+sum([Jan^income],...  
[Dec^income])

RTN

Enters the formula for Total^Income.

Total^Income: 24000.00

BUDGET:	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Total
Income:	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	24000.00
House :	400.00	400.00	400.00	400.00	400.00	400.00							
Insur :	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	
Taxes :	600.00	600.00	600.00	600.00	600.00	600.00							
Food :	200.00	175.00	180.00	155.00	195.00	140.00							
Util :	110.00	95.00	87.00	81.00	68.00	55.00							
TotExp:	1410.00	1370.00	1367.00	1336.00	1363.00	1295.00							
Remain:	590.00	630.00	633.00	664.00	637.00	705.00							

Now, let's create another row beneath the Remain row called Exp(%). This row will contain the calculation for the total monthly expenses (TotExp) as a percentage of the total annual income. We will use the following formula:  $Exp(\%) = TotExp \div Total\ Income * 100$ .

Input/Result

SHIFT > [Jan^remain] RTN

Goes to cell Jan^Remain.

Jan^Remain: 590.00

↓

Goes to the next row.

[Jan]11:

/ [H] [R] Exp(%) RTN ATTN

Enters a new row header and returns to Top level.

Jan^Exp(%):

```
[Jan^totexp]/[total^income]  
#100 RTN
```

Enters a formula for cell Jan^Exp(%).

```
Jan^Exp(%) 5.88
```

Now replicate the Exp(%) row across the worksheet. In the Exp(%) formula, the value for TotExp is different each month, and that variable should have a relative replication. However, since the annual income does not change, the total income variable should be copied with no change.

**Input/Result**

```
/ R
```

Executes the Replicate command.

```
Repl source: [Jan^Exp(%)]
```

```
RTN
```

Defines the source range.

```
Repl target: 
```

Prompts for the target range.

```
[feb^exp(%)][...][dec^exp(%)]
```

```
RTN
```

Displays the replicate options.

```
Repl: No change Relative Both
```

```
B
```

Selects both types of replication.

```
Jan^Exp(%) [Jan^TotExp] +No Rel
```

Jan^TotExp is a cell reference for which the no change (N) or relative (R) option is available.

```
R
```

Selects relative replication for the Jan^TotExp reference.

```
Jan^Exp(%) [Total^Incom] +No Rel
```

```
N
```

Selects the no change option for the Total^Income reference.

```
Jan^Exp(%) 5.88
```

BUDGET:	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Total
Income:	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	24000.00
House :	400.00	400.00	400.00	400.00	400.00	400.00							
Insur :	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	
Taxes :	600.00	600.00	600.00	600.00	600.00	600.00							
Food :	200.00	175.00	180.00	155.00	195.00	140.00							
Util :	110.00	95.00	87.00	81.00	68.00	55.00							
TotExp:	1410.00	1370.00	1367.00	1336.00	1363.00	1295.00							
Remain:	590.00	630.00	633.00	664.00	637.00	705.00							
Exp(%):	5.88	5.71	5.70	5.57	5.68	5.40	.00	.00	.00	.00	.00	.00	

The area of the display that contains the formula (between the cell coordinates field and the `+No Rel` prompt) can be viewed the same way that a formula is viewed at Top level (with `TAB`, `SHIFT TAB`, `RUN`, or `SHIFT RUN`).

The `TIME` key, to toggle the cell coordinates field, is also active during the `+No Rel` prompt. `SHIFT TIME`, which toggles the cell contents field, is active only when each new formula in the source range is displayed for the first time.

## Other Types of Replication

In the examples so far, you have been replicating a single cell into another cell or into a range of cells. You can also replicate an entire column into another column, and replicate a column across a range of adjacent columns. The same is true for rows.

### Replicating a Range of Cells

Replicating a range of cells is similar to the replications you have seen so far. You simply set the cell display to the first cell whose contents are to be replicated, and press `/ R` (or press `/ R CLR` and type in the coordinates of the first cell of the source range). VisiCalc prompts `Repl source:` and the coordinates of the current cell. You type an ellipsis, enter the cell coordinates of the last cell of the source range, and press `RTN`. VisiCalc then prompts for the target range—that is, the cells into which the contents of the source range are to be repeated. You enter the first and last cells of the target range, separated by an ellipsis. When you are replicating a column or row into another column or row, the target range should consist of the cells that will receive a copy of the first cell of the source column or row.



If an expression in the source range is a formula with a cell reference, you will also have to specify whether the formula is to be repeated with no changes, relative to its new location, or both.

**Example:** In the BUDGET1 worksheet, let's create a second monthly budget beneath the budget that we have been using. The first step is to replicate the values from the Jan column (cells Jan^Income to Jan^Remain) into cells A14 to A23, the area on the worksheet that we have designated for the new budget (the Exp(%) row will not be replicated). Formulas should be replicated relative to other formulas. (We could create new row headers, but we will not do so at this time.)

### Input/Result

[SHIFT] > [Jan^Income] [RTN]

Goes to the first cell in the source range.

[/] [R]

Executes the Replicate command.

Repl source: [Jan^Income] ■

Prompts for the source range.

...[Jan^Remain] [RTN]

Enters values in the Jan column as the source range.

Repl target: ■

Prompts for the target range.

A14 [RTN]

Enters values in the target range. The Jan column will be copied, beginning at cell A14 and continuing downward.

Repl: No change Relative Both

Displays the replicate options.

[R]

Selects the relative option.

Jan^Income: 2000.00

BUDGET:	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Total
Income:	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	24000.00
House :	400.00	400.00	400.00	400.00	400.00	400.00							
Insur :	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	
Taxes :	600.00	600.00	600.00	600.00	600.00	600.00							
Food :	200.00	175.00	180.00	155.00	175.00	140.00							
Util :	110.00	95.00	87.00	81.00	68.00	55.00							
TotExp:	1410.00	1370.00	1367.00	1336.00	1363.00	1295.00							
Remain:	590.00	630.00	633.00	664.00	637.00	705.00							
Exp(%):	5.88	5.71	5.70	5.57	5.68	5.40	.00	.00	.00	.00	.00	.00	
	2000.00												
	400.00												
	100.00												
	600.00												
	200.00												
	110.00												
	-----												
	1410.00												
	590.00												

Replicating Into a Range of Columns or Rows

In order to replicate a column of cells across several columns, simply specify as the source range the top and bottom cells of the column, and specify as the target range the top cells of the first and last columns into which the column is to be repeated.

Likewise, to replicate a row of cells down several rows, simply specify the left and right cells of the row as the source range, and the left-most cells of the top and bottom rows into which the row is to be repeated as the target range.

If the cells in the source range contain cell references, you will also have to specify whether the formula is to be repeated with no changes, relative to the new location in the worksheet, or both.

**Example:** Now let's replicate cells A14 through A23 across the entire worksheet (columns Feb through Dec).

### Input/Result

a14

Goes to cell A14.

[Jan]14: 2000.00

Executes the Replicate command.

Repl source: [Jan]14■

Prompts for the source range.

...[Jan]23

Enters [Jan]14 through [Jan]23 as the source range.

Repl target: ■

Prompts for the target range.

[Feb]14...[Dec]14

Enters the target range.

Repl: No change Relative Both

Displays the replicate options.

Selects the relative option.

[Jan]14: 2000.00

Notice that while we split this example into two parts, the entire process could have been done using [Jan^Income]...[Jan^Remain] as the source range and [Jan]14...[Dec]14 as the target range. Replications involving larger source and target ranges could take substantially longer.

BUDGET:	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Total
Income:	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	24000.00
House :	400.00	400.00	400.00	400.00	400.00	400.00							
Insur :	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	
Taxes :	600.00	600.00	600.00	600.00	600.00	600.00							
Food :	200.00	175.00	180.00	155.00	195.00	140.00							
Util :	110.00	95.00	87.00	81.00	68.00	55.00							
TotExp:	1410.00	1370.00	1367.00	1336.00	1363.00	1295.00							
Remain:	590.00	630.00	633.00	664.00	637.00	705.00							
Exp(%):	5.88	5.71	5.70	5.57	5.68	5.40	.00	.00	.00	.00	.00	.00	
	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00	
	400.00	400.00	400.00	400.00	400.00	400.00	400.00	400.00	400.00	400.00	400.00	400.00	
	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	
	600.00	600.00	600.00	600.00	600.00	600.00	600.00	600.00	600.00	600.00	600.00	600.00	
	200.00	200.00	200.00	200.00	200.00	200.00	200.00	200.00	200.00	200.00	200.00	200.00	
	110.00	110.00	110.00	110.00	110.00	110.00	110.00	110.00	110.00	110.00	110.00	110.00	
	1410.00	1410.00	1410.00	1410.00	1410.00	1410.00	1410.00	1410.00	1410.00	1410.00	1410.00	1410.00	
	590.00	590.00	590.00	590.00	590.00	590.00	590.00	590.00	590.00	590.00	590.00	590.00	

As you have seen, an entire worksheet can be created with one or two Replicate commands, making Replicate one of the most powerful and useful VisiCalc commands. As you practice and become more familiar with Replicate, you will find that it can save you literally hours that you would otherwise spend typing in values, formulas, and labels.

# **Part II**

## **Using VisiCalc<sup>®</sup> With Peripheral Devices**

**Part II** provides information on expanding the use of VisiCalc with HP-75 compatible peripherals.

**Part II** contains the following sections:

- **Section 5** (“Storing and Retrieving Worksheet Files”) explains how you can store your worksheet files on magnetic cards or other mass storage media.
- **Section 6** (“Using a Printer with VisiCalc”) describes how to use a peripheral printer to print your worksheets. In this section you will learn how to connect and assign a peripheral printer, print a worksheet with or without headers, and print formulas.
- **Section 7** (“Using a Video Display with VisiCalc”) describes how to use a video display device to display your worksheets. In this section you will learn how to install and assign the video interface and a video display device, activate and deactivate the video display, and interpret the information presented by the video worksheet display.
- **Section 8** (“Formatting the Worksheet”) describes how to alter the appearance of the worksheet on a video display or on a printer. In this section you will learn how to use the Local and Global format commands, change the column width, and use Local Graphing format.

## Storing and Retrieving Worksheet Files

### Contents

What You'll Learn in This Section .....	103
Using Magnetic Cards .....	103
Using a Mass Storage Device .....	104

### What You'll Learn in This Section

Since worksheet files are simply another HP-75 file type, the mass storage operations built in to the HP-75 are used to store and retrieve worksheet files to and from mass storage devices like the card reader and the HP 82161A Digital Cassette Drive. Because card reader operations and mass storage operations are fully described in the *HP-75 Owner's Manual*, this section will only cover those aspects of mass storage operations that are unique to worksheet files. Before reading this section, you should have read and understood the information in section 1.

### Using Magnetic Cards

You can use the HP-75's built-in card reader to transfer worksheet files between memory and magnetic cards. Worksheet files can be copied from memory to magnetic cards or from magnetic cards to memory. In either case, a file transfer results in duplicate files—one in memory and one on a magnetic card or cards. The use of magnetic cards is fully described in section 8 of your *HP-75 Owner's Manual*, "Card Reader Operations." If you are not familiar with magnetic card use, please read section 8 of the owner's manual before attempting card reader operations.

All aspects of card reader operations covered in section 8 of the *HP-75 Owner's Manual* apply to worksheet files (type `W` files). You should, however, note the following information:

- The `PCRD` device code is not valid for worksheet files. (It can only be used with BASIC program files.)
- Since worksheet files can't be `EDITed` (that is, they can never be the current edit file), you must always specify a worksheet file name when copying worksheet files to cards.
- It is *not* necessary to have the VisiCalc module installed in your HP-75 when transferring worksheet files between memory and cards—in either direction.
- Worksheet card files can be `PROTECTED` and `UNPROTECTED` just like any other card file.



## Using a Mass Storage Device

The HP-75's built-in HP-IL interface enables you to store and retrieve worksheet files on a mass storage medium other than magnetic cards. Just as with magnetic cards, worksheet file-transfers to and from mass storage result in duplicate files, with one copy on the mass storage medium and one in memory. Mass storage operations are fully described in section 9 of the *HP-75 Owner's Manual* under the heading "Mass Storage Operations." If you are not familiar with mass storage operations, please read section 9 before attempting mass storage operations.

**Note:** While you can store and retrieve worksheet files from the card reader without having the VisiCalc module installed, the module must be in place in the computer to copy worksheet files to or from an HP-IL mass storage medium. If you attempt to store or retrieve worksheet files from an HP-IL mass storage medium without the module installed, the HP-75 will display error 68—`w r o n g   f i l e   t y p e`—and cancel the copy operation. Mass storage worksheet files can, however, be `CATalogued`, `RENAMED`, and `PURGED` even if the module is not installed.

## Using a Printer With VisiCalc®

### Contents

What You'll Learn in This Section .....	105
Connecting and Assigning the Printer .....	105
Using Other Printers .....	106
Using the Print Command .....	106
Specifying the Print Boundaries .....	109
Controlling the Printer .....	110
Specifying the Printer Width .....	112
Printing the Worksheet Formulas .....	115
The Effect of Formats and Column Widths on Printed Worksheets .....	117

### What You'll Learn in This Section

In this section you will learn how to use a peripheral printer and VisiCalc to produce printed copies of your worksheets. When you finish this section, you should be able to:

- Connect and assign a peripheral printer on the HP-75's Hewlett-Packard Interface Loop (HP-IL).
- Use the **/P** (*Print*) command to print worksheets with or without the column and row headers.
- Use the **/P** command to make a printed list of the formulas used in a worksheet.
- Use the `Printer setup:` and the `Printer width:` prompts of the **/P** command to control printer operation and title your worksheet printouts.

Before reading this section, you should have read and understood the information in sections 1, 2, 3, and 4.

### Connecting and Assigning the Printer

Before you can use the **/P** command to print a worksheet, you must correctly connect an HP-IL compatible printer to the HP-75 and assign it a device code. In this section, we will use the HP 82905B Opt. 248 Impact Printer to illustrate printer operations.

The procedure for connecting and assigning an HP-IL compatible printer is covered in detail in section 9, “HP-IL Operations,” of the *HP-75 Owner’s Manual*. Please refer to that section for information if you are unfamiliar with HP-IL operations.

The procedure below briefly summarizes, for your reference, the steps necessary to prepare the printer for use with VisiCalc.

To connect and assign a printer on the loop, first make sure that the HP-75 is turned off, then:

1. Ready the printer according to the installation instructions in the owner’s manual for the printer.
2. Attach the printer to the loop.
3. Turn on the printer and any other devices on the loop.
4. Turn on the HP-75 and execute an `ASSIGN IO` command to assign a device code to the printer and to reassign device codes to existing peripherals on the loop.
5. Execute a `PRINTER IS` command to declare the assigned printer as the current print device.

After you complete this procedure, the printer will be ready to accept printed output from VisiCalc.

## Using Other Printers

With VisiCalc and the HP-75 computer, you can use any peripheral printer that is compatible with HP-IL. The VisiCalc Print command enables you to set the width of the printed output to match the printer you are using. In addition, `/P` allows you to send *control codes* to the printer that can modify the way the worksheet is printed. The owner’s manual for your printer will indicate what control codes the printer will respond to and how to specify them.

## Using the Print Command

The Print command enables you to print an entire VisiCalc worksheet or only a specified portion of a worksheet. You also have the option of printing the worksheet either with or without headers. Another option enables you to print a list of the formulas used in the worksheet.

To use the print command, press `[/]` to enter Command level, then press `[P]`. VisiCalc will prompt:

```
Pr: Worksheet +Headers Formulas
```

The *print options menu* displays the three Print command options.

Select one of the three options by pressing the appropriate letter key.

**/PW** *Worksheet*. Prints the worksheet without column or row headers.

**/PH** *+Headers*. Prints the worksheet with column and row headers.

**/PF** *Formulas*. Prints a list of the formulas used in the worksheet. The print formulas option will be discussed on page 115.

The principal difference between the **/PW** and **/PH** options is illustrated by the sample worksheet printouts below. As you can see, the Worksheet printout consists of the values (and labels, if any) that are contained in the worksheet cells. The +Headers printout contains, in addition, both the default and user-defined column and row headers of the worksheet along with lines to separate the headers from the actual worksheet area.

**Note:** In the remainder of this section, the concepts presented will be illustrated using the original **BUDGET** worksheet, which was stored on the magnetic card packaged with VisiCalc. To produce the same printouts as those shown in the examples that follow, copy the original **BUDGET** worksheet file into memory. (You may have to purge or rename the current **BUDGET** worksheet file to do this.)

BUDGET	A	B	C	D	E	F		BUDGET
	Jan	Feb	Mar	Apr	May	Jun		
1 Income:	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00		2000.00 2000.00 2000.00 2000.00 2000.00 2000.00
2								
3 House :	400.00	400.00	400.00	400.00	400.00	400.00		400.00 400.00 400.00 400.00 400.00 400.00
4 Insur :	100.00	100.00	100.00	100.00	100.00	100.00		100.00 100.00 100.00 100.00 100.00 100.00
5 Taxes :	600.00	600.00	600.00	600.00	600.00	600.00		600.00 600.00 600.00 600.00 600.00 600.00
6 Food :	200.00	175.00	180.00	155.00	195.00	140.00		200.00 175.00 180.00 155.00 195.00 140.00
7 Util :	110.00	95.00	87.00	81.00	68.00	55.00		110.00 95.00 87.00 81.00 68.00 55.00
8								
9 TotExp:	1410.00	1370.00	1367.00	1336.00	1363.00	1295.00		1410.00 1370.00 1367.00 1336.00 1363.00 1295.00
10 Remain:	590.00	630.00	633.00	664.00	637.00	705.00		590.00 630.00 633.00 664.00 637.00 705.00

You can use the **/HSY** command (introduced in section 2) to suppress the printing of *default* headers in the **/PH** option of the Print command. To do this, you would execute the Suppress Header command, as described below, before executing the Print command.

**Input/Result**

☐ / Enters Command level.

Command: DEGHIMPEVW-

☐ H Initiates the Header command.

Headers: Col Row Width Suppress

S	Selects the suppress headers prompt.
---	--------------------------------------

```
Suppress default headers: Yes No
```

Y	Executes the command. Default headers will now be suppressed in worksheet printouts (also in video displays—discussed in the next section).
---	---------------------------------------------------------------------------------------------------------------------------------------------

If you had executed the Suppress Header command before executing the Print command, the worksheet-plus-headers printout would look like this:

BUDGET:	Jan	Feb	Mar	Apr	May	Jun
Income:	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00
House :	400.00	400.00	400.00	400.00	400.00	400.00
Insur :	100.00	100.00	100.00	100.00	100.00	100.00
Taxes :	600.00	600.00	600.00	600.00	600.00	600.00
Food :	200.00	175.00	180.00	155.00	195.00	140.00
Util :	110.00	95.00	87.00	81.00	68.00	55.00
TotExp:	1410.00	1370.00	1367.00	1336.00	1363.00	1295.00
Remain:	590.00	630.00	633.00	664.00	637.00	705.00

You can use the **/HW** (*Header Width*) command to suppress the printing of *user-defined* headers. If you execute the Header Width command and set the row header width to 0, user-defined headers will not be printed by the worksheet-plus-headers option of the Print command. If default headers have also been suppressed (by **/HSY**), the resulting printout won't have *any* headers—it will look just like the one produced by the print worksheet option. If default headers are not suppressed (**/HSN** is in effect), then only default headers will be printed with the worksheet, as illustrated below:

## BUDGET

	A	B	C	D	E	F
1:	2000.00	2000.00	2000.00	2000.00	2000.00	2000.00
2:						
3:	400.00	400.00	400.00	400.00	400.00	400.00
4:	100.00	100.00	100.00	100.00	100.00	100.00
5:	600.00	600.00	600.00	600.00	600.00	600.00
6:	200.00	175.00	180.00	155.00	195.00	140.00
7:	110.00	95.00	87.00	81.00	68.00	55.00
8:						
9:	1410.00	1370.00	1367.00	1336.00	1363.00	1295.00
10:	590.00	630.00	633.00	664.00	637.00	705.00

## Specifying the Print Boundaries

After you press **[W]**, **[H]**, or **[F]** to select the *type* of printout, you must select the *area* of the worksheet to be printed. To accomplish this, VisiCalc prompts for the coordinates of the upper-left and lower-right cells to be printed. The first prompt is:

Upper left: *cell coordinates*

Every time the Print command is executed, VisiCalc displays the coordinates of the upper-left cell of the worksheet (cell **A1**). To specify that printing should begin from this cell, press **[RTN]**. To specify a different upper-left cell, press **[CLR]**, type the coordinates of the desired boundary cell, then press **[RTN]**. You can enter user-defined, default, or mixed coordinates—VisiCalc will display the coordinates in this and subsequent prompts according to the current setting of the **[TIME]** key toggle.

After the upper-left boundary is specified, VisiCalc prompts for the lower-right print boundary:

Lower right: *cell coordinates*

Every time the Print command is executed, VisiCalc displays the coordinates of the cell at the lower-right boundary of the *active* worksheet area. To use the default boundary, press **[RTN]**. To specify a new lower-right cell, press **[CLR]**, type the new coordinates, then press **[RTN]**. As with the upper-left boundary, you can enter user-defined, default, or mixed coordinates.

## Controlling the Printer

The two cell coordinates specified in the *upper-left* and *lower-right* prompts define a rectangular area of the worksheet that will be printed by the **/P** command. This area may include the entire worksheet (and will if the default boundaries are used), or it may include only a part of the worksheet. In either case, the number of characters per line needed to print the specified area of the worksheet may exceed the number of characters that the printer can place on a line.

For example, the HP 82905B printer normally prints 80 characters per line. A worksheet with a global column width\* of 8 and with 12 worksheet columns would require a 96-character print line to fit one *row* of the worksheet (excluding headers) on one *line* of the printout. Since you may often need to print worksheets that need more characters than the available printer width, the VisiCalc Print command has been designed so that it produces usable printouts regardless of printer line-length limitations. The features of the Print command that make this possible are controlled by the last two prompts of the Print command: the `Printer setup:` prompt and the `Printer Width:` prompt.

After you specify the print boundaries, VisiCalc prompts for an optional *printer setup string*:

`Printer setup: current printer setup string`

You can control the operation of most printers by sending them various *control codes* and/or *escape sequences*. The `Printer setup:` prompt provides you with a way to conveniently transmit these codes to the printer. You can place any valid control code, escape sequence, or combination into the printer setup string. For example, with the HP 82905B printer, you could place the *form-feed* control code (`(CTL)(L)`) into the setup string to cause the printer to feed to the beginning of a new page before the worksheet is printed. For a description of the codes that are valid for your printer, refer to the owner's manual for the printer.

Some printers—including the HP 82905B—have the ability to print several different sizes of characters. For example, the optional printer setup string can be used to tell the HP 82905B printer to change from printing standard sized characters to *compressed* characters—132 of which will fit on a line. With that line length, longer worksheet rows can fit on one line of the printout.

Compressed print is initiated on the HP 82905B by placing an *escape sequence* in the setup string. Escape sequences begin with the escape character (ESC—ASCII character code 27), followed by one or more other characters. To place the escape character in the setup string press the `(CTL)(BACK)` key-stroke. The ESC character is represented in the display by the greater-than symbol, `>`.

---

\* Global column width is described in section 8, "Formatting the Worksheet " (page 139).



**Example:** Place the escape sequence for “compressed print” in the printer setup string.

### Input/Result

**CTL** **BACK** &k2S

Before you press **RTN**, VisiCalc shows the entire string.

Printer setup: >&k2S■

**RTN**

After you press return, VisiCalc displays the final prompt, the Printer width: prompt (discussed beginning on page 112).

Printer width: 32

Printer setup strings are stored with the worksheet file, so whenever you print a worksheet for which a printer setup string has previously been specified, the setup string will be displayed with the printer setup prompt. If you press **RTN** *without altering the string*, that same string will be sent to the printer. If you wish to change the setup string, you must press the **CLR** key and enter the new string from the beginning. You can’t alter a setup string by editing one or more characters; if you attempt to do so, the ESC character will be lost from the string and the string will no longer be an escape sequence.

**Note:** Because of the way the HP-75 display handles the ESC character, it will appear in the display only at the time you actually enter it into the setup string. If you recall the setup string to the display by executing the Print command again, the escape symbol and the next character of the escape sequence will not appear in the display—although they are still maintained in memory as part of the setup string.

Other escape sequences can be placed in the setup string in a similar fashion. Escape sequences are described in the owner’s manual for the printer. Note that it is not necessary to put quotes around printer setup strings.

In addition to control codes and escape sequences, printer setup strings can contain printable characters that you can use to title or annotate worksheet printouts. For example, the following setup string would cause the printer to print the title “Budget printout for first three months.” before the worksheet. Both the title and the worksheet would be printed in compressed characters.

**CTL** **BACK** &k2SBudget printout for first three months.

You can put up to 95 characters in a printer setup string, including escape and control characters. As long as they do not contain the ESC character, you can edit setup strings using the HP-75 editing keys. Remember that setup strings which *do* contain the ESC character must be cleared and reentered if changes are required.

## Specifying the Printer Width

The final prompt of the Print command is the `Printer Width:` prompt. This prompt asks you to tell VisiCalc how many characters per line your printer can print. The first time you use the Print command with a particular worksheet, this prompt displays a default width equal to the current HP-75 `PWIDTH` setting:

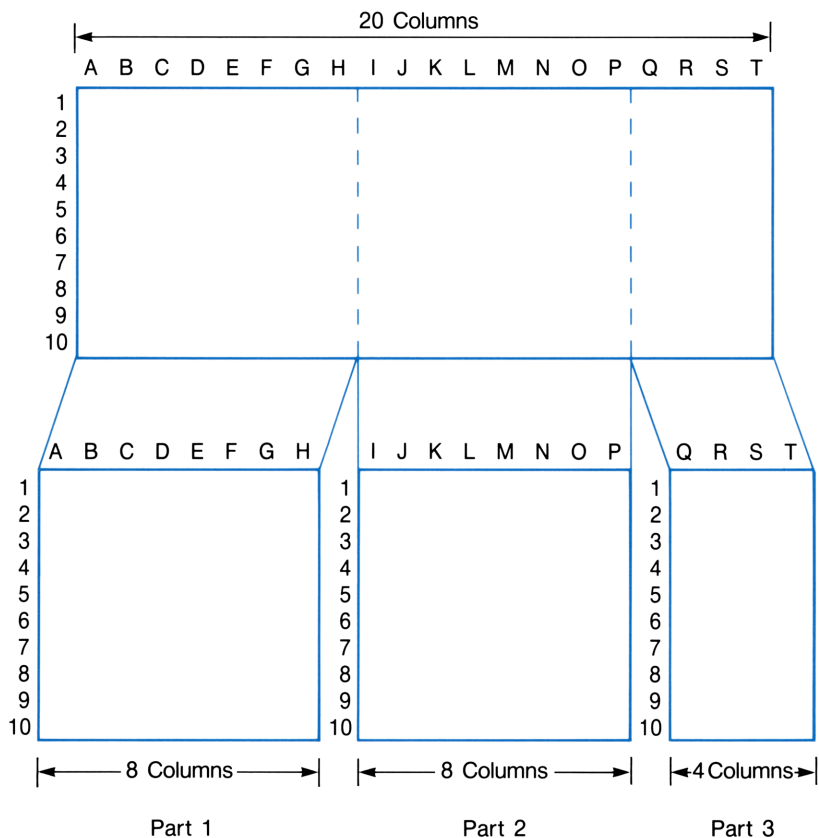
`Printer width: current HP-75 PWIDTH setting`

If you press `[RTN]` at this point, VisiCalc will begin printing the worksheet according to the specifications you have given in the preceding Print command prompts. If you want to specify a different printer width, type a new printer width over the displayed width and press `[RTN]`—VisiCalc will now use the new printer width (the HP-75's `PWIDTH` will also be set to this new value).

**Note:** As with all VisiCalc commands, you can terminate the Print command (and return to Top level) by pressing the `[ATTN]` key at any step in the process. Even after the printer begins printing, pressing `[ATTN]` will stop the printer and return VisiCalc to Top level. (You may have to hold the `[ATTN]` key down for a second or two before the printer will stop.)

The `Printer width:` prompt is the key to the other feature that allows VisiCalc to handle worksheets wider than the print line. Remember that by using compressed characters on the HP 82905B, you can print 132 characters on a line. Even this extended width is insufficient for many worksheets, however, so VisiCalc uses the specified printer width to handle worksheets that are too large to fit on a particular printer's line width—whatever it is. VisiCalc does this by calculating how many worksheet columns will fit on the specified printer width. When a worksheet is printed, VisiCalc prints the worksheet in parts. Each part contains only as many worksheet columns as will fit in the specified printer width (and as many rows as specified by the boundary prompts in the Print command). After the first part is printed, subsequent parts begin where the previous part left off and again fill the available printer width.

**Example:** Consider a worksheet with 20 columns and with 9 characters in each cell (illustrated below). To print one row of that worksheet (20 columns  $\times$  9 characters/cell) on a single line of the printout, a print line of 180 characters would be required. Suppose your printer has an 80 character line length (printer width = 80) and you want to print the entire worksheet. Obviously one row of the worksheet will not fit on one line of the printout. Instead, because of its ability to calculate how many columns will fit on the available printer width, VisiCalc would print the worksheet in parts. Each part would consist of the column headers for the columns in that part, the row headers, and only as many columns as would fit in the available printer width. The illustration below shows how the printout would be divided by VisiCalc for the conditions in this example.



If you specify the worksheet-plus-headers option (/PH), VisiCalc will print the row headers at the beginning of each part. The number of character positions required to print the row headers depends on the current row header width and on whether default headers have been suppressed. Default headers require up to 4 character positions and the user-defined header width can be set to any value between 0 and 255, so the number of columns of the worksheet that are printed per part may be reduced. The example worksheet `BUDGET` is printed below using a 32-character printer width. The first printout was done with the worksheet-only option; the second was done with the worksheet-plus-headers option and with a header width of 6. Note that the version with headers required three parts to print, while the worksheet-only version required only two parts.

BUDGET			
2000.00	2000.00	2000.00	2000.00
400.00	400.00	400.00	400.00
100.00	100.00	100.00	100.00
600.00	600.00	600.00	600.00
200.00	175.00	180.00	155.00
110.00	95.00	87.00	81.00
-----			
1410.00	1370.00	1367.00	1336.00
590.00	630.00	633.00	664.00
2000.00	2000.00		
400.00	400.00		
100.00	100.00		
600.00	600.00		
195.00	140.00		
68.00	55.00		
-----			
1363.00	1295.00		
637.00	705.00		

BUDGET	A	B
	Jan	Feb
-----+-----		
1 Income	2000.00	2000.00
2		
3 House	400.00	400.00
4 Insur	100.00	100.00
5 Taxes	600.00	600.00
6 Food	200.00	175.00
7 Util	110.00	95.00
8	-----	
9 TotExp	1410.00	1370.00
10 Remain	590.00	630.00

BUDGET	C	D
	Mar	Apr
-----+-----		
1 Income	2000.00	2000.00
2		
3 House	400.00	400.00
4 Insur	100.00	100.00
5 Taxes	600.00	600.00
6 Food	180.00	155.00
7 Util	87.00	81.00
8	-----	
9 TotExp	1367.00	1336.00
10 Remain	633.00	664.00

BUDGET	E	F
	May	Jun
-----+-----		
1 Income	2000.00	2000.00
2		
3 House	400.00	400.00
4 Insur	100.00	100.00
5 Taxes	600.00	600.00
6 Food	195.00	140.00
7 Util	68.00	55.00
8	-----	
9 TotExp	1363.00	1295.00
10 Remain	637.00	705.00

Because of the way that VisiCalc handles printer widths, you don't need to be concerned about whether your worksheet printouts will be confusing or even unusable because the worksheet won't fit in the available printer width. Simply specify the printer width of your printer and VisiCalc will take care of formatting the output in the most usable form. Even if you use the battery operated HP 82162A Thermal Printer, with its 24-character print line, VisiCalc will produce readable printouts of your worksheet—as long as your worksheet column width is 24 characters or less.

## Printing the Worksheet Formulas

As mentioned at the beginning of this section, VisiCalc allows you to make a printed list of the formulas used in the worksheet. When you select the **F**ormula option of the **P**rint command, the remainder of the **P**rint command prompts appear in the same order as previously described for the print worksheet options. Using the boundary prompts, you can print all or only some of the formulas for the worksheet. The **P**rinter **s**etup: prompt will display any string that was previously entered for that worksheet; and the **P**rinter **w**idth: prompt will display the current HP-75 **P**WIDTH setting.

The **/PF** printout consists of a list, in column order, of the coordinates and contents of each cell in the worksheet that contains a formula. The listing duplicates the cell display for each cell as it appears in Formula Display mode. Cells with label- or value-only entries are not included in this listing. The formula listing for the worksheet printed at the beginning of this section (page 107) might look, in part, like this:

```
BUDGET

Jan^House: .2*[Jan^Income]
Jan^Taxes: .3*[Jan^Income]
Jan^TotExp: SUM([Jan^House]...[Jan^Util])
Jan^Remain: [Jan^Income]-[Jan^TotExp]

Feb^Income: [Jan^Income]
Feb^House: .2*[Feb^Income]
Feb^Taxes: .3*[Feb^Income]
Feb^TotExp: SUM([Feb^House]...[Feb^Util])
Feb^Remain: [Feb^Income]-[Feb^TotExp]

Mar^Income: [Feb^Income]
Mar^House: .2*[Mar^Income]
Mar^Taxes: .3*[Mar^Income]
Mar^TotExp: SUM([Mar^House]...[Mar^Util])
Mar^Remain: [Mar^Income]-[Mar^TotExp]
```

In this case the formulas are printed with user-defined headers. Like the cell display, the formula listing will follow the current settings of the `[TIME]` key and the `[SHIFT][TIME]` keystroke. For example if you have pressed the `[TIME]` key to toggle the cell coordinates field of the cell display to default headers, the formula listing above would look like this :

```
BUDGET

A3: .2*[Jan^Income]
A5: .3*[Jan^Income]
A9: SUM([Jan^House]...[Jan^Util])
A10: [Jan^Income]-[Jan^TotExp]

B1: [Jan^Income]
B3: .2*[Feb^Income]
B5: .3*[Feb^Income]
B9: SUM([Feb^House]...[Feb^Util])
B10: [Feb^Income]-[Feb^TotExp]

C1: [Feb^Income]
C3: .2*[Mar^Income]
C5: .3*[Mar^Income]
C9: SUM([Mar^House]...[Mar^Util])
C10: [Mar^Income]-[Mar^TotExp]
```

By using various combinations of the `[TIME]` and `[SHIFT][TIME]` toggles, you have complete control over the way cell coordinates are printed in formula listings. By changing one or both fields of the formula listing to default headers you can print longer formulas on one line when using a printer with a restricted line length—like the HP 82162A Thermal Printer. If the formulas *are* too long to fit on one line, VisiCalc will print the cell coordinates on one line, then the formula on the next, like this:

```
BUDGET

Jan^House:
.2*[Jan^Income]
Jan^Taxes:
.3*[Jan^Income]
Jan^TotExp:
SUM([Jan^House]...[Jan^U
til])
Jan^Remain:
[Jan^Income]-[Jan^TotExp
]
```

The print formulas option provides a useful reference list of the formulas in a worksheet. For example, if you replicate a formula in a cell across a row, you could print the formulas for that entire row to check that the replication was what you expected.

## The Effect of Formats and Column Widths on Printed Worksheets

Another concept that is important to your understanding and use of the Print command is worksheet *format*. VisiCalc provides several formatting commands that enable you to completely control the appearance of your printed worksheets (and of the video display). Formatting was mentioned briefly in section 2 under “The Display Format Toggle” (page 63) and will be discussed in detail in section 8, “Formatting the Worksheet.”

Earlier in this section (page 107) you saw how to use the Suppress Header command (**/HS**) and the Header Width command (**/HW**) to control the printing of default and user-defined headers. In section 8 you will learn how to use the various formatting commands to control the appearance of printed numbers and labels (**/F** and **/GF**), and to control the width of worksheet columns (**/W** and **/GC**). Please refer to section 8 if you want more information about formatting worksheet printouts at this time.

Under certain conditions one row of a worksheet printout may wrap around to a second line of the printout. Recall that VisiCalc normally prints the worksheet in parts. This means that the contents of each cell will be printed on one line and not be split between lines if the cell happens to fall at the end of the printed line. However, if one or more of the columns in the area to be printed has a column width that is too wide to fit—along with the row headers—in the current printer width, then a cell may be split between lines.

For example, suppose the current printer width is 80, a column to be printed has width 80 also, and the worksheet is to be printed with default headers. When the worksheet is printed, VisiCalc will print the worksheet row headers (4 character positions), plus the first 76 characters of the cell contents, on one line. The remaining 4 characters of the cell contents will then be printed at the beginning of the next line. The next row in that column of the worksheet will then begin on the next printed line and wrap to a second line, and so on.

Any column of a worksheet that is too wide to fit in the current printer width will be printed as a separate part containing only the headers and the rows in that column. Also, if the user-defined *row* header width is greater than the current printer width, the user-defined headers will be suppressed and only the default headers will be printed (unless they have also been suppressed by the **/HSY** command).





## Using a Video Display With VisiCalc®

### Contents

What You'll Learn in This Section .....	119
Connecting and Assigning the Display .....	119
Activating and Deactivating the Video Display .....	120
Using the Video Display .....	121
How the Video Display Affects VisiCalc Operation .....	122
Interpreting the Video Display .....	123
The Effect of Formats and Column Widths on the Video Display .....	128

### What You'll Learn in This Section

In this section you will learn how to use an HP-IL compatible video interface with a video monitor or a television set to display worksheets. A video interface provides you with a bigger “window” into the worksheet—a display window that may contain many worksheet cells. When you finish this section, you should be able to:

- Correctly install and assign the video interface and a video display device.
- Turn the video display off and on, and understand why you might wish to do so.
- Change the way that VisiCalc shows the header fields of the video worksheet display. Convert the video display to any of the four possible header display options.
- Interpret the way information is presented by the video worksheet display, and use the video worksheet display to enhance the power of VisiCalc.

Before reading this section, you should have read and understood the information in sections 1, 2, 3, and 4.

### Connecting and Assigning the Display

To use a video display device with VisiCalc, you must first connect and assign the video display device and the video interface. The video interface connects to the Hewlett-Packard Interface Loop, and the display device (either a video monitor or a television set) connects to the interface. Instructions for

connecting a video display to the video interface can be found in the owner's manual for the video interface. The procedure for connecting and assigning a video interface to the loop is contained in section 9, "HP-IL Operations," of the *HP-75 Owner's Manual*. Please refer to that section for information if you are unfamiliar with HP-IL operations.

The procedure below briefly outlines, for your reference, the steps necessary to prepare a video interface for use with VisiCalc.

1. Connect the interface to a source of ac power and to a video display device according to the installation instructions in the owner's manual for the video interface.
2. Attach the video interface to the loop.
3. Turn on all of the devices on the loop.
4. Turn on the HP-75 and execute an `ASSIGN IO` command to assign a device code to the video interface and to reassign device codes to existing peripherals on the loop.
5. Execute a `DISPLAY IS` command to declare the assigned video interface as a display device.

When the VisiCalc module is first installed, the VisiCalc program is set to produce a video display with 16 lines of 32 characters each. If your video interface has different display length and width limits, you can change the default settings of VisiCalc to the values appropriate for your interface. The procedure for doing this is described in appendix E of this manual, "Setting the Video Display Parameters" (page 186). Check the owner's manual for your display interface for information about the display parameters appropriate for your interface, then refer to appendix E if you need to change the default video display parameters.

**Note:** The video displays in this and the following section represent the displays produced by the HP 82163 Video Interface. If your video interface and display device provide different display parameters, your displays may show more (or less) of the example worksheets than is shown here.

After you complete this procedure, the video interface will be ready to display the contents of the active VisiCalc worksheet.

## Activating and Deactivating the Video Display

With the video display connected and initialized, VisiCalc will automatically display the portion of the worksheet that will fit into the display limits of the video interface. Every time the worksheet is recalculated or the cell display is moved to a part of the worksheet not currently visible on the video screen, VisiCalc "writes" a new display to the video screen. Because VisiCalc requires more time to write the worksheet to the video display than to the HP-75 display window, there are times when it would be convenient to be able to turn the video display off—without having to completely disconnect it.

VisiCalc provides this capability by means of the **/V** (*Video*) command. To turn the video display off, press ☐ **/V**. VisiCalc responds with the following prompt:

```
Video active: Yes No
```

- /VY** If the video was previously deactivated by a **/VN** command, this option will first clear any existing display, then reactivate the video display. If the video is already active when this option is executed, then the display is cleared and immediately rewritten.
- /VN** If the display was previously activated by a **/VY** command, this option turns the cell cursor off but leaves the current display intact. While the cell cursor is off, changes to the worksheet will not get written to the display. **/VN** has no effect if the video is already inactive.

After a **/VN** command, the video display will remain inactive until **/VY** is executed or until you exit and restart VisiCalc. (VisiCalc executes an automatic **/VY** at start-up.)

Regardless of the length of the video display line produced by your interface, VisiCalc displays a line with one less character than the maximum so as to avoid the automatic scrolling that occurs when a character is written to the last character position of the last screen line. For example, if your video interface produces a 32 character display line, VisiCalc will display a maximum line length of 31 character. The number of characters actually displayed depends on the column width and number of columns in the worksheet.

Under certain conditions extraneous characters may get written to the video screen. When this happens you can clear those characters, without affecting the rest of the worksheet display, by executing a **/VY** command.

## Using the Video Display

Although VisiCalc was designed to work effectively with the single line display window of the HP-75, adding a video display to the system enhances your use of VisiCalc by providing a larger view of the worksheet.

The operation of a video display with VisiCalc is almost totally “transparent.” Other than connecting and assigning the video interface and the video display device, you don’t really have to *do* anything to use the video display. You do, however, need to know what effect the display has on the normal operation of VisiCalc, and how to interpret and format the information presented by the video display.

## How the Video Display Affects VisiCalc Operation

As mentioned under “Activating and Deactivating the Video Display,” VisiCalc requires time to rewrite the worksheet to the display—in fact, several seconds are required for a large and complex worksheet. During certain types of operations, such as when you are entering a series of cell values, you may not wish to wait for VisiCalc to rewrite the display each time you terminate cell entry. You can avoid these delays by using the **/VN** command to deactivate the video display.

There are several other effects of using a video display with VisiCalc:

- The video display provides a much larger window into the worksheet. This makes it easier to keep track of the relationship between various elements of the worksheet.
- The video display always shows the *result* of any cell formula, regardless of the current display mode. Switching the cell display to Formula Display mode (by pressing **[APPT]**), allows you to see both *formulas* (in the HP-75 display) and *results* (in the video display) as you work.
- Like worksheet printouts, the video display can show both user-defined and default headers for each column and row. This means that you can conveniently use either type of header to reference cells. You can cause the video display to show only user-defined headers (by executing the **/HSY** command described on page 107 in section 6) or you can cause the display to show only default headers (by setting the header width to 0 with the **/HW** command).
- As with the Print command, you can use the **/F** (*Local Format*) and **/GF** (*Global Format*) commands to completely control the way labels and numbers are displayed.\* (Only the Integer and Dollars formats have any effect on the HP-75 cell display; all of the other Format commands affect only the video display and the printer.)
- You can use the **/W** (*Local Column Width*) command to specify separate column widths for each column in the worksheet, the **/HW** (*Header Width*) command to control the row header width, and the **/GC** (*Global Column Width*) command to set the global column width.\*
- The video display gives you a preview of the way a formatted worksheet will appear in printed form.

Except for the slight delays that occur when the video display is rewritten, and the other effects listed above, the video display has no other effect on the operation of VisiCalc.

**Note:** In the remainder of this section, the concepts presented will be illustrated using the original **BUDGET** worksheet, which was stored on the magnetic card packaged with VisiCalc. To see the same displays as those shown in the examples that follow, copy the original **BUDGET** worksheet file into memory. (You may have to purge or rename the current **BUDGET** worksheet file to do this.)

---

\* The **/F**, **/GF**, **/W**, **/HW**, and **/GC** commands are described in the next section, page 131.



BUDGET	Jan	Feb	Mar
Income	2000.00	2000.00	2000.00
House	400.00	400.00	400.00
Insur	100.00	100.00	100.00
Taxes	600.00	600.00	600.00
Food	200.00	175.00	180.00
Util	110.00	95.00	87.00
TotExp	1410.00	1370.00	1367.00
Remain	590.00	630.00	633.00

The worksheet is displayed with user-defined headers. Note that another worksheet column becomes visible because the removal of the default headers provides four more display character positions, enough so that another worksheet column will fit in the display.

If you execute the **/HW** command and set the header width to zero, all headers will disappear and only the worksheet cells will appear in the display.

**Input/Result**

☐ ☐ H ☐ W

Width row headers: 6

☐ RTN

Working

VisiCalc displays the row header width prompt.

2000.00	2000.00	2000.00
400.00	400.00	400.00
100.00	100.00	100.00
600.00	600.00	600.00
200.00	175.00	180.00
110.00	95.00	87.00
-----		
1410.00	1370.00	1367.00
590.00	630.00	633.00

Finally, if you turn off default header suppression by executing the **/HSN** command, you will get a display that contains default headers and the cell entries.

☐ / ☐ H ☐ S

Suppress default headers: Yes No

VisiCalc displays the suppress default headers prompt.

☐ N

Working

	A	B	C
1	2000.00	2000.00	2000.00
2			
3	400.00	400.00	400.00
4	100.00	100.00	100.00
5	600.00	600.00	600.00
6	200.00	175.00	180.00
7	110.00	95.00	87.00
8	-----		
9	1410.00	1370.00	1367.00
10	590.00	630.00	633.00
11			
12			
13			
14			
15			

Note that when the user-defined headers are suppressed as in the video display above, another row of the worksheet becomes visible at the bottom of the display.



The four possible video display combinations that result from various settings of the Header Width and Default Header Suppress commands are summarized below:

	/HWn (n ≠ 0)	/HW0
/HSN	The full video display (page 123).	Default headers only, with no worksheet name (page 125).
/HSY	User-defined headers only (page 124).	No headers at all. Only the cell entries are shown (page 125).

As you can see above, the video worksheet display contains a lot of information. To use this information effectively, we need to introduce a new concept: the cell cursor. When you use VisiCalc in a portable environment, your “window” into the worksheet is a single cell. Throughout this manual the term “cell display” has been used to describe this single-cell window. When the video display is active, many cells are displayed on the screen at once and we need an indicator to show which is the “current” cell. The *cell cursor* is that indicator; it is located at cell Jan^Income in the illustrations above.

The cell marked by the cell cursor in the *video display* is the same cell that is shown in the *cell display* of the HP-75. If you press one of the arrow keys to move the cell display to another cell, the cell cursor on the video display moves to the corresponding cell. With the video display active, when you move the cell *display* you also move the the cell *cursor*.

For example, with the cell display and cell cursor at cell Jan^Income:

**Input/Result**

[ / ] [ H ] [ W ]

Width row headers: 0

VisiCalc displays the header width prompt.

6 [ RTN ]

Resets the header width to 6 characters.


[ APPT ]

Sets BUDGET to *Formula Display mode*.

Jan^Income: 2000.00

Cell display.

BUDGET		A	B
		Jan	Feb
1	Income	2000.00	2000.000
2			
3	House	400.00	400.000
4	Insur	100.00	100.000

Now, if you press the right arrow key, , both the cell display and the cell cursor will move to cell Feb^Income:

**Input/Result**



Feb^Income: [Jan^Income]

BUDGET		A	B
		Jan	Feb
1	Income	2000.00	2000.00
2			
3	House	400.00	400.00
4	Insur	100.00	100.00

Since Formula Display mode is in effect, the cell display will show the formula stored in cell Feb^Income, and the video display will show the value that is the result of that formula.

If you move the cell cursor beyond the boundaries of the video display window, VisiCalc will rewrite the display. For example:

**Input/Result**



Mar^Income: [Feb^Income]

Moves the cursor to cell Mar^Income.

BUDGET		B	C
		Feb	Mar
1	Income	2000.00	2000.00
2			
3	House	400.00	400.00
4	Insur	100.00	100.00

The cursor has moved to Mar^Income and the Jan column has scrolled off the left side of the display. Notice that the row header field remains fixed on the screen.

Both of the header fields remain fixed in the video display at all times, although the headers move with the corresponding row or column. As the cell cursor is moved to cells that are outside of the right or lower boundaries of the display, columns and rows at the left and top of the worksheet move off the screen behind the header fields. Since the column and row fields remain fixed in the display, you always know what part of the worksheet you are looking at.

If you make a cell entry that causes the worksheet to be recalculated, there will be a short delay while the recalculation occurs and the video display is rewritten. VisiCalc will display *Working* during this delay, and the cell cursor will be turned off. If you are entering a lot of interrelated data and want to avoid the delays that occur during recalculation, you can turn off the video display as described previously, or you can set VisiCalc to Manual Recalculation mode by executing the */GRM* (*Global Recalculation Manual*) command (page 80).

## The Effect of Formats and Column Widths on the Video Display

As with printed output, the VisiCalc formatting commands (*/F* and */GF*), the column width commands (*/W* and */GC*), and the Header command (*/H*) give you complete control over the appearance of the video display. In this section you learned how to use the */HS* and */HW* commands to control the display of worksheet headers. The next section, “Formatting the Worksheet,” provides complete information about using the */F*, */GF*, */W*, and */GC* commands.

If you specify worksheet column widths or a row header width greater than the available video display width, VisiCalc tries to fit columns into the display in the most logical way. The process by which VisiCalc attempts to fit columns into the display is as follows:

1. First VisiCalc checks the row header width. If it is less than the video display width, then the user-defined row headers are written to the display—otherwise only the default headers (if they haven’t been suppressed) are displayed.
2. VisiCalc begins placing columns in the display, beginning with the first column that will fit in the remaining video display width.
3. Subsequent columns are written to the display, if they will also fit in the video display width, until a column is reached that won’t fit.

4. If the column containing the current cell is *not* displayed (because it won't fit or because it follows a column that won't fit), then no cell cursor will appear. However, the current cell remains visible in the HP-75 display window.
5. The columns that *are* displayed will always be ones that are adjacent to one another in the worksheet. For example, if column E won't fit, column C will never be displayed next to column H.



## Formatting the Worksheet

### Contents

What You'll Learn in This Section .....	<b>131</b>
Local Formatting .....	<b>132</b>
Available Formats .....	<b>133</b>
Replicating a Local Format .....	<b>135</b>
Local Column Width .....	<b>136</b>
Global Formatting .....	<b>138</b>
Available Formats .....	<b>138</b>
Global Column Width .....	<b>139</b>
Local Graphing Format .....	<b>140</b>
Overflow Display .....	<b>140</b>

### What You'll Learn in This Section

This section explains how to alter the appearance of cell contents on a video display or on a printer. When you finish this section, you should be able to:

- Know the Local format commands and what each one does.
- Replicate a Local format.
- Set local column width.
- Know the Global format commands and what each one does.
- Use Local Graphing format.
- Know what the overflow display looks like.

Before reading this section, you should have read and understood the information in sections 1, 2, 3, 4, and 6 or 7.

Remember that formatting, whether done globally or locally, alters only how the numbers and labels on the VisiCalc worksheet appear on a video display or printer. Internally, each cell can hold a label of up to 95 characters or a formula with 91 characters, and VisiCalc always computes cell values to the full precision of the HP-75.

**Note:** The formatting of your VisiCalc worksheet is possible only when a video display device and/or a printer are attached. Except for Integer and Dollars formats, the formatting commands have no effect on the way a value or number is displayed in the HP-75's display window. For illustration purposes, this section assumes that a video display is being used; however, formatting will work the same way on a printer.

## Local Formatting

When you first create a worksheet, all cells in the worksheet default to the global format. In this format, left and right justification have no effect on the way a value or label is displayed. However, when viewed on the video display, labels are left-justified and values are right-justified. Numbers are displayed in whatever form will show the most information about each entry.

You can alter the format of a selected cell by specifying the **/F** (*Local Format*) command.

To change the format of a selected cell:

1. Move the cell cursor to the desired cell. (Remember, the video display should be connected.)
2. Press **/** **F**. VisiCalc prompts:

```
Format (local):  D G I L R $ *
```

3. Select the desired local format by pressing the letter key for that format. The format is executed immediately, and the appearance of the number or label is changed to the new local format.

The local format defined for a cell overrides the global format.

## Available Formats

Here are the specific formats abbreviated with the **/F** command.

**/FD** *Local Default to Global format.* This command causes the cell entry to default to the current global format.

**/FG** *Local General format.* With this command, values in the current cell are right-justified with as many digits displayed as possible. Labels in the current cell are left-justified beginning with the left-hand column. This lets you select the same format for a cell that was set when the worksheet was first created, regardless of the global format over the rest of the worksheet.

**/FI** *Local Integer format.* With this command, values in the current cell are rounded to an integer and are right-justified without a decimal point.

**/FL** *Local Left-Justified format.* With this command, values or labels in the current cell are left-justified.

**/FR** *Local Right-Justified format.* With this command, values or labels in the current cell are right-justified.

**/F\$** *Local Dollars format.* With this command, values in the current cell are rounded to two decimal places and are right-justified.

**/F\*** *Local Graphing format.* This command converts cell values to asterisks. For example, a value of 5 (or -5) returns five asterisks (\*\*\*\*\*). Fractional values are rounded, not truncated.

The type of local format appears in the local format field of the status line.

**Example:** Let's create a new worksheet called **FORMAT** and examine the number 1.7 in several formats. Exit VisiCalc (if necessary), create the new worksheet, then use the following keystrokes.

### Input/Result

A1:

Displays cell A1 of the new worksheet.

1.7 RTN

Enters the value 1.7 in cell A1.

A1: 1.7

/ R

Executes the Replicate command.

Repl source: A1■

Prompts for the source range.



RTN

Enters cell A1 as the source range.

Repl target: ■

Prompts for the target range.

B1...F1 RTN

Enters cells B1 through F1 as the target range.

FORMAT	A	B	C
1	1.7	1.7	1.7
2			
3			

We now have the same number, 1.7, replicated in six positions in row 1. Now we'll change the formats of these numbers, beginning with the contents of cell E1.

Input/Result

→ / F I

Selects Integer format.

FORMAT	A	B	C
1	1.7	2	1.7
2			
3			

As you can see, the appearance of the number in cell E1 has been changed to Integer format. The number is rounded, with none of its decimal places visible in the display. Naturally, it maintains its full value internally, and you can see this value by pressing SHIFT APPT.

Let's try some different formats on the other numbers.

**Input/Result**

→ / F L

Moves to cell C1 and selects Left-Justified format.

→ / F R

Moves to cell D1 and selects Right-Justified format.

FORMAT	B	C	D
1		2 1.7	1.7
2			
3			

**Input/Result**

→ / F \$

Moves to cell E1 and selects Dollars format.

→ / F \*

Moves to cell F1 and selects Local Graphing format.

FORMAT	D	E	F
1	1.7	1.70	**
2			
3			

Local formatting always takes precedence over the global format. The only way to change or clear a local format is with the /F command.

**Replicating a Local Format**



Just as you can replicate the label, value, or formula in a cell down columns or across rows (replication was discussed in section 4), you can also replicate the format. In fact, whenever you replicate a cell, if a local format has been specified in that cell, the local format is replicated as well. Each new cell appears in that specific local format.

To replicate *only* a local format, just specify the format with the /F command, and clear the contents of the cell. (If the cell is already blank, just use the /F command to select its format.) Then replicate the cell over the desired range with the /R command. All cells replicated in the range will be in the selected format, ready for data.

## Local Column Width

Another feature of VisiCalc is the ability to specify a local column width with the **/W** (*Width*) command. With this command, each column of a worksheet can have a different width—anywhere from 0 to 255 characters.

To use the Width command:

1. Move the cell display to any cell in the column.
2. Press  . VisiCalc prompts:

Width of col    *current column coordinate: current local column width*

3. Key in the desired column width and press **RTN**. To reset to the global column width, press **CLR** **RTN**.

**Example:** Add the following numbers to the `FORMAT` worksheet. Then use the `/W` command to specify that column A has width 2 and that column B has width 9.

	A	B
1	2	123456
2	3	12345678
3	5	1234

### Input/Result

SHIFT ←

Goes to cell A1.

A1: 1.7

2 RTN ↓

3 RTN ↓

RTN

Stores the values in column A.

SHIFT   ↑   →

Goes to cell E1.

B1: 2

123456 

RTN

↓

12345678 

RTN

↓

1234 

RTN

Stores the values in column B.

FORMAT	A	B	C
1	2	123456	1.7
2	3	1.23E7	
3	5	1234	
4			

Input/Result

⌈ W

Width of col B: █

9 

RTN

⬅

⌈ W

Width of col A: █

2 

RTN

Executes the Width command.

Sets the width of column B to 9.

Goes to column A.

Executes the Width command.

Sets the width of column A to 2.

FORMAT	A	B	C
1	2	123456	1.7
2	3	12345678	
3	5	1234	
4			

There are several points to remember when using the **/W** command.

- The **/W** command alters only the *appearance* of the cells on the video display or the printer; it does not change their contents.
- The local column width has precedence over the global column width (page 139). That is, if a local column width has been specified for a column, the column will be printed or displayed with that width, not the current global column width.
- If a local column width is specified for a particular column, the width will remain with that column, even if it is moved, or if other columns around it are inserted, deleted, or moved.
- If a local column width has been specified for column **IU** (the right-most possible column in the worksheet), VisiCalc interprets this condition as if the worksheet contained 255 columns and you will not be able to insert any more columns with **/I**.

## Global Formatting

The **/G** (*Global*) command enables you to change the width of all of the columns on the worksheet at once, to change the format over the entire worksheet, and to select recalculation mode and order options. (Unlike the formatting commands, recalculation *can* alter the actual contents of the cells. Recalculation was discussed in section 3.)

### Available Formats

When you select the Global command by pressing **/G**, VisiCalc prompts:

```
Global: Col Order Recalc Format
```

The available Global commands are:

**/GC** *Global Column Width*. VisiCalc prompts:

```
Col width (global): 7
```

This command sets the column width to any width from 0 to 255 characters. The default column width is seven characters.

**/GO** *Global Order of Recalculation*. VisiCalc prompts:

```
Order of recalc: Col Row
```

The order of recalculation of a new worksheet is column-by-column.

**/GR** *Global Recalculation mode*. VisiCalc prompts:

```
Recalc: Auto Manual
```

Automatic Recalculation mode is set when a worksheet is first created.

**/GF** *Global* format. VisiCalc prompts:

```
Format (global): G I L R £ ¢
```

This command sets the format of all cells in the worksheet except those that have had a local format set with the **/F** command. The available global formats are the same as the available local formats, except that **/GFD** is not available.

Refer to section 3 (page 80) for a discussion of the Recalculation commands **/GO** and **/GR**.

### Global Column Width

The **/GC** (*Global Column Width*) command enables you to change the column width of the VisiCalc worksheet so that the cells are displayed with anywhere from 0 to 255 characters. To select a global column width, type **/** **G** **C** followed by the desired width.

The **/GC** command affects only the *appearance* of the cells that you examine when you view the worksheet (either with a video display or a printer); it does not change their contents.

**Example:** In the **FORMAT** worksheet, change the global column width to four characters.

#### Input/Result

**/** **G** **C**

Executes the Global Column Width command.

```
Col width (global): 7
```

4 **RTN**

Sets column width to 4 characters. Notice that column **D** is now displayed.

FORMAT	A	B	C	D
1	2	123456	1.7	1.7
2	3	12345678		
3	5	1234		
4				

Like the **/W** command, the Global Column Width command alters only the *appearance* of the cells on the video display or the printer; it does not change their contents. Also remember that the local column width has precedence over the global column width. That is, if a local column width has been specified for a column, the column will be printed or displayed with that width, not the current global column width.

## Local Graphing Format

The **/F\*** (*Local Graphing format*) command displays the number of asterisks specified by the value in the selected cell. As many asterisks as possible will be displayed, depending on the cell's width. If the value in the cell is not an integer, the value is rounded to the nearest integer. If the value is negative, its absolute value is used.

This command can be used, with a video display or a printer, to create bar graphs of functions and values in VisiCalc. As with all **/F** commands, **/F\*** is a local format. To change the format of more than one cell at a time, you can use the **/R** (*Replicate*) command (described on page 135) or the **/GF\*** command.

## Overflow Display

For all formats except Dollars format, as the column width is decreased the display of values is first rounded (if there is a fraction). The display then goes to scientific notation, and then to an “overflow” display that uses the “greater than” (>) symbol. Here are some examples of overflow displays:

Column Width	Value	/FG	/FI	/F\$	/F*
7	106496	106496	106496	>>>>>	*****
6	106496	1.1E5	1.1E5	>>>>>	*****
5	106496	1E5	1E5	>>>>	****
4	106496	1E5	1E5	>>>	***
3	106496	>>	>>	>>	**
7	12.3586	12.359	12	12.36	*****
6	12.3586	12.36	12	12.36	*****
5	12.3586	12.4	12	>>>>	****
4	12.3586	12	12	>>>	***
3	12.3586	12	12	>>	**

Notice that although rounding, scientific notation, and overflow occur on the video display, they do not occur on the HP-75's display in Unformatted Display mode. When Formatted Display mode is set, rounding and scientific notation alter the HP-75's display in Integer format. In Dollars format, only rounding alters the HP-75's display.\*

Even though a cell may overflow in the video display, the actual cell contents remain unaltered within VisiCalc. To see the actual cell contents, look at the HP-75's display, and use the SHIFT APPT key-stroke combination if Formatted Display mode is in effect.

\* There are some numbers, such as 1E12, that cannot be displayed as dollars. In these cases, full precision will be shown in the HP-75's display and overflow will show on the video display.

# **Part III**

## **Advanced VisiCalc® Operations**



**Part III** describes two advanced features that greatly expand the power and flexibility of HP-75 VisiCalc. Because of the advanced nature of this material, you should be thoroughly familiar with VisiCalc before you attempt to use the features described in part III.

**Part III** contains the following sections:

- **Section 9** (“Accessing Another Worksheet From a Cell”) describes how to use an external cell reference to access a cell in another worksheet from a cell in the current worksheet.
- **Section 10** (“Extending the VisiCalc Function Set”) describes how to extend the VisiCalc function set by creating and using extension functions.

## Accessing Another Worksheet From a Cell

### Contents

What You'll Learn in This Section .....	143
External Cell References .....	143
Simplifying a Complex Worksheet .....	144
Adding a Third Dimension .....	146
Specifying a Single Cell in Another Worksheet .....	148
Specifying a Range of Cells in Another Worksheet .....	150
Inconsistent Worksheets With External References .....	150

### What You'll Learn in This Section

In this section you will learn how to use an *external cell reference* to access a cell in another worksheet from a cell in the current worksheet. When you finish this section, you should be able to:

- Describe the two general situations in which it might be desirable to create multiple worksheets to solve a particular problem.
- Correctly enter an external cell reference into a cell.

Before reading this section, you should have read and understood the information in sections 1, 2, and 3.

### External Cell References

A powerful feature of VisiCalc is the ability to use *external cell references*—that is, the ability to reference another worksheet in memory directly from a cell. Placed in a cell in one worksheet, an external cell reference accesses information from one or more cells in another worksheet.

There are two general situations when you might want to use multiple worksheets linked by external cell references:

- To simplify a complex, multi-part worksheet so that each part is more efficient and easier to use.
- To provide a “third dimension” to worksheets. This capability is useful whenever you want to extend the analysis obtained in a worksheet through another dimension—such as a budget for one period over multiple periods, or a cost analysis for one project over an entire range of projects.

## Simplifying a Complex Worksheet

Suppose that you are developing a worksheet to evaluate a piece of income property. One part of the worksheet contains pertinent information about the property: mortgages, income, depreciation, expenses, etc. Another part of the worksheet contains the actual year-by-year analysis of the property showing cash flows, returns, etc. The worksheet might look like this:

PROPERTY	A	B	C
1	Variables		
2			
3 Mortgage	80000.00		
4 Term	30.00		
5 Interest	12.25		
6 Payment	838.32		
7			
8 Price	100000.00		
9 Apprec	6.00		
10 Income	9900.00		
11 Expenses	3300.00		
12 Deprec	125.00		
13 Tax %	45.00		
14 Gain Tax	30.00		
15			
16	Analysis		
17			
18	Year 1	Year 2	Year 3
19			
20 NOI	6600.00	6600.00	6600.00
21 -Int	9784.90	9749.27	9709.01
22 -Deprec	4687.50	4394.53	4119.87
23 =Tax Inc	-7872.40	-7543.80	-7228.88
24 #Tax %	45.00	45.00	45.00
25 =Taxes	-3542.58	-3394.71	-3253.00
26 Net Inc	6600.00	6600.00	6600.00
27 -Pmts	10059.84	10059.84	10059.84
28 -Taxes	-3542.58	-3394.71	-3253.00
29 =ATCF	82.74	-65.13	-206.84
30			
31 Sale	106000.00	112360.00	119101.60
32 -Costs	6360.00	6741.60	7146.10
33 -Bal	79725.06	79414.49	79063.66
34 -Tot Tax	1438.88	4647.43	7840.01
35 +ATCF	82.74	-65.13	-206.84
36 =Tot CF	18558.80	21491.35	24844.99
37			
38 Eq Yield	-7.21	3.87	7.54

As you can see, this worksheet is quite large, and it contains a number of empty cells. In this case, you could make more efficient use of memory space (remember that empty cells use memory), and you could find your way around the worksheet(s) easier, if the single worksheet above were divided into two worksheets. One worksheet could contain the pertinent property data, and the other worksheet could contain the year-by-year analysis.

The two worksheets below contain the same information as the single worksheet above. Note how restructuring the problem in this way has reduced the number of required cells and has also made the user-defined headers for each cell meaningful in all cases. Because new headers were created, it is possible to refer to cell Year 3^Eq Yield instead of cell CCEq Yield].

VARIABLE	A	ANALYSIS	A	B	C
			Year 1	Year 2	Year 3
1 Mortgage	80000.00	1 NOI	6600.00	6600.00	6600.00
2 Term	30.00	2 -Int	9784.90	9749.27	9709.01
3 Interest	12.25	3 -Deprec	4687.50	4394.53	4119.87
4 Payment	838.32	4 =Tax Inc	-7872.40	-7543.80	-7228.88
5	-----	5 *Tax %	45.00	45.00	45.00
6 Price	100000.00	6 =Taxes	-3542.58	-3394.71	-3253.00
7 Apprec	6.00	7 Net Inc	6600.00	6600.00	6600.00
8 Income	9900.00	8 -Pmts	10059.84	10059.84	10059.84
9 Expenses	3300.00	9 -Taxes	-3542.58	-3394.71	-3253.00
10 Deprec	125.00	10 =ATCF	82.74	-65.13	-206.84
11 Tax %	45.00	11	-----	-----	-----
12 Gain Tax	30.00	12 Sale	106000.00	112360.00	119101.60
		13 -Costs	6360.00	6741.60	7146.10
		14 -Bal	79725.06	79414.49	79063.66
		15 -Tot Tax	1438.88	4647.43	7840.01
		16 +ATCF	82.74	-65.13	-206.84
		17 =Tot CF	18558.80	21491.35	24844.99
		18	-----	-----	-----
		19 Eq Yield	-7.21	3.87	7.54

If you were to look at a formula printout for the PROPERTY worksheet on the previous page, you would find that numerous references were made, in the "Analysis" part of the worksheet, to cells in the "Variables" part of the worksheet. By using external cell references, we are able to make these same references in the VARIABLE and ANALYSIS worksheets above. The references, in this case, are between cells in different worksheets instead of within the same worksheet.

You can decide if it would be worthwhile dividing one worksheet into several less complex ones by reviewing the following list of questions. If you can answer “yes” to one or more of these questions, then multiple worksheets, linked by external cell references, should be considered:

- Does your proposed worksheet contain many unused cells that would be eliminated if the worksheet were divided? (In the `PROPERTY` worksheet, cells `B1` through `B14` and cells `C1` through `C14` are empty.)
- If you are planning to use user-defined headers to refer to cells in your worksheet, do some of the user-defined cell coordinates refer to cells that should have different labels than those specified by the coordinates? (In the `PROPERTY` worksheet, cell `BCNOI` would be better defined as `Year 2^NOI`.)
- Does your proposed worksheet break down naturally into several related parts, such as an input data section, a computation section, and a result section? (In the `PROPERTY` worksheet, cells `ACMortgage` through `ACGain Tax` represent the property data, and the remaining cells represent the property analysis computation.)
- Does your proposed worksheet contain one part—such as an input data part—that changes often, while other parts of the worksheet change relatively little? Remember that it takes less time for VisiCalc to recalculate a small worksheet than a larger one.

## Adding a Third Dimension

Suppose the `COSTS` worksheet (described in section 2) is one of several worksheets containing information on individual manufacturing projects, and you would like to combine some of the information from each worksheet onto a new worksheet called `TOTAL`, which might look like this:

Worksheet TOTAL			
TOTAL	Proj 1	Proj 2	Proj 3
Q1 Cost			
Q2 Cost			
Tot Cost			
Q1 Units			
Q2 Units			
# Units			

The information for Project 1 is in the `COSTS` worksheet, which currently looks like this:

<code>COSTS</code>	A	B
	Qtr 1	Qtr 2
1 Labor	48400	49600
2 Material	56700	58900
3 Tot Cost		
4 # Units	11000	14500
5 Unit Cost	9.5545455	7.48

The `Q1 Cost` (quarter 1 cost) item in worksheet `TOTAL` is the total cost from quarter 1 of `COSTS`.<sup>\*</sup> The `Q2 Cost` (quarter 2 cost) item in worksheet `TOTAL` is the total cost from quarter 2 of `COSTS`. Likewise, the `Q1 Units` (quarter 1 number of units) item in `TOTAL` is the number of units in quarter 1 of `COSTS`. The `Q2 Units` item is the number of units in quarter 2 of `COSTS`.

`COSTS` worksheets for several projects can be created, each containing information specific to a single project. These separate `COSTS` worksheets could be maintained and updated by separate project managers. As changes occur in any of the `COSTS` worksheets, the `TOTAL` worksheet is automatically updated by means of the external cell references it contains. At any time, the manufacturing manager could call up the `TOTAL` worksheet to see the summary of the status of all the projects.

These simple examples only begin to explore the possibilities inherent in the external cell reference concept. As you learn to use external references, you will find a whole range of new analytical possibilities.

---

<sup>\*</sup> We did not actually enter formulas into the `Tot Cost` row. However, the formula would be the sum of the labor and material costs in each quarter.

## Specifying a Single Cell in Another Worksheet

In its simplest form, an external cell reference specifies a single cell in another worksheet. To enter an external cell reference in a cell:

1. Move the cell display to the cell where the reference should be placed.
2. Type in the cell coordinates of the external cell, followed by a colon. The external cell can be referenced using user-defined coordinates, default coordinates, or a combination of the two.
3. Type in the name of the worksheet in which the cell resides (this worksheet must reside in memory).
4. Press **[RTN]**.

When VisiCalc encounters an external cell reference during worksheet recalculation, the value associated with the external cell is placed in the cell containing the external cell reference.

External cell references can be used in formulas, in the parameter lists of VisiCalc functions such as **SUM** and **MEAN**, as parameters for HP-75 numeric functions such as **SIN** or **INT**, and in the parameter lists of extension functions (described in section 10).

The following examples illustrate two types of external cell references:

```
A2: A1+C5:TAXES
```

Adds the value found in cell C5 of worksheet TAXES to cell A1 of the current worksheet, and stores the result in cell A2.

```
A5: SUM(D3:WEEK1,D3:WEEK2,D3:WEE
```

Computes the sum of cell D3 in worksheets WEEK1, WEEK2, etc., and stores the result in cell A5.



**Example:** Create the worksheet `TOTAL` and the user-defined headers shown on page 146, then use the following keystrokes to specify the appropriate information in the `COSTS` worksheet. (The `COSTS` worksheet must reside in memory. Refer to page 147 for its current values.)

### Input/Result

[SHIFT] [>] [Proj 1^q1 units] [RTN]

Moves the cell display to cell  
Proj 1^Q1 Units.

Proj 1^Q1 Units:

[Qtr 1^# units]:costs [RTN]

Enters in cell Proj 1^Q1 Units an external  
cell reference to the cell in the `COSTS`  
worksheet.

Proj 1^Q1 Units\_ 11000

[↓] [Qtr 2^# units]:costs [RTN]

Enters in cell Proj 1^Q2 Units an external  
cell reference.

Proj 1^Q2 Units\_ 14500

Similar references could be placed in the remainder of column `Proj 1`, and references to other worksheets could be placed in columns `Proj 2` and `Proj 3` to complete the `TOTAL` worksheet.

If you try to create an external cell reference specifying a worksheet that does not reside in memory, the HP-75 will beep and place the cursor on the first character of the worksheet name.

If you recalculate a worksheet that contains external cell references to a worksheet that is not in the HP-75's memory, each cell in the worksheet containing an external cell reference will have the value `NA`. This means that the referenced worksheet is not available. The external cell reference formula remains intact. If the formula is displayed, it will be shown with default headers.

When a column or row is deleted, inserted, moved, or replicated in a worksheet, external cell references to that worksheet, from other worksheets, are not updated to reflect the new coordinates.

## Specifying a Range of Cells in Another Worksheet

It is also possible for an external cell reference to specify a range of cells in another worksheet in memory. The cells in the range are specified, followed by a colon and the name of the worksheet in which the cells reside (this worksheet must reside in memory). As before, the external cells can be referenced using user-defined coordinates, default coordinates, or a combination of the two.

**Example:** In the previous example, suppose that the `COSTS` worksheet had been expanded to include quarterly information for a full year. In that case, the total number of units of project 1 would be the sum of the number of units from cells `Qtr 1^# Units` to `Qtr 4^# Units` of the `COSTS` worksheet. The cells could be accessed in the following manner:

```
+sum([Qtr 1^# units]...[Qtr 4^# units]:costs)
```

## Inconsistent Worksheets With External References

When using external cell references, it is possible to create *inconsistent worksheets*. An inconsistent worksheet occurs when the worksheet’s calculated results do not reflect the current data in the worksheet. Let’s use the following worksheets to illustrate this concept.

TEST	A	B
1		
2		A2: DATA
3		

DATA	A
1	
2	SUM(A1...A3:TEST)
3	

If cells `A1`, `A2`, and `A3` change in the current worksheet `TEST`, then the `DATA` worksheet must be recalculated before `B2` in `TEST` will be correct. `TEST` is therefore an inconsistent worksheet.

VisiCalc has a solution to help alleviate this potential problem. When a worksheet contains one or more external cell references, and Automatic Recalculation mode is set (with **/GRA**), the worksheet will be recalculated at startup, before you get to Top level. However, only the active worksheet is recalculated, so it is still possible to have an inconsistent worksheet in memory.

An inconsistent worksheet can also occur if a forward or circular reference is created between worksheets. Refer to “Forward and Circular References” in section 3 for a complete description of this phenomenon.

## Extending the VisiCalc® Function Set

### Contents

What You'll Learn in This Section .....	151
What Is an Extension Function? .....	152
Accessing an Extension Function From a Cell .....	153
Passing Values To and From Extension Functions .....	155
Creating Your Own Extension Functions .....	159
Input/Output Operations .....	159
Trapping Input Errors .....	161
Putting Together a Model .....	161
Avoiding and Correcting Design Errors .....	164
LEX Files in RAM or ROM .....	166

### What You'll Learn in This Section

In this section you will learn how to extend the VisiCalc function set by creating and using *extension functions*. When you finish this section, you should be able to:

- Understand what an extension function is, and how it can be used to provide additional computational capability for VisiCalc.
- Enter a valid extension function call in a worksheet cell.
- Understand how VisiCalc passes parameters to extension functions through the `VISIDATA` file.
- Write simple extension functions that correctly access `VISIDATA` to read parameters and return a result.

Before reading this section, you should have read and understood the information in sections 1, 2, and 3. In addition, before you read “Passing Values To and From Extension Functions” (page 155) and “Creating Your Own Extension Functions” (page 159), you should be familiar with the information presented in part III, “Programming the HP-75,” of the *HP-75 Owner's Manual*.

## What Is an Extension Function?

In section 1 we introduced the idea that VisiCalc allows you to extend the VisiCalc function set by creating your own *extension functions*. In this section, we are going to take a closer look at what extension functions are and how they work. Extension functions are created by writing a BASIC program that has some special characteristics. These characteristics, which are shared by VisiCalc functions and HP-75 numeric functions, are listed below:

- The function has zero or more numeric *parameters* (inputs). HP-75 numeric functions have a fixed number of parameters, VisiCalc functions (like `SUM`) can have as many parameters as will fit in the entry line, and extension functions can have up to 9,999 parameters.
- The function performs a numeric computation on the parameters. For example, `SUM` computes the sum of its parameters; the HP-75 function `SIN` computes the sine of its single parameter.
- The function returns a single numeric *result*. All of the functions available to VisiCalc—including extension functions—return their result to the calling cell.

Extension functions, like other functions accessible through VisiCalc, can't handle string data—either as input parameters, or as a result. Remember, however, that if a parameter list contains a label or an empty cell, VisiCalc passes the numeric value 0 for that cell to the function.

These characteristics limit this class of functions to the manipulation of numbers, but even within this limit, extension functions provide the means to solve very complex problems.

Extension functions can expand the function set of VisiCalc to include functions that perform computations in specialized areas of knowledge. For example, a set of extension functions designed to compute various financial values, like IRR (internal rate of return) or NPV (net present value), could make it possible to perform a complete range of complex financial calculations in a worksheet.

A magnetic card that contains an extension function called `IRR` was packaged with your VisiCalc product. This function was included to demonstrate the power of extension functions, as well as to provide an example of how extension functions are designed. The `IRR` function computes an internal rate of return of a series of ungrouped cash flows, and returns the IRR as a *decimal* value to the calling cell.\* In this section we will use the `IRR` function to illustrate the use and design of extension functions.

---

\* The `IRR` function computes a *unique, positive* internal rate of return. It will return `ERROR` to the calling cell if more than one IRR can be computed from the inputs or if no positive IRR exists. Also, the `IRR` function requires numeric parameters, so it will return `ERROR` if any of the parameters passed to it are labels.

Find the card that contains the `IRR` function and load it into memory by executing a copy-from-card command. The complete procedure for copying a file from a magnetic card to memory can be found in section 8, “Card Reader Operations,” of the *HP-75 Owner's Manual*. Please refer to that section if you are unfamiliar with card reader operations.

After you complete the copy-from-card command, the `IRR` function can be called from any worksheet.

## Accessing an Extension Function From a Cell

As mentioned in section 1, calls to extension functions look exactly like calls to VisiCalc or HP-75 functions—the only difference is that the function name is that of a BASIC file stored in memory. For example, to place a call to the `IRR` function in a cell, the cell entry might look like this:

```
A11(V): @irr(a1...a10)■
```

Like a built-in function call, a call to an extension function consists of a function name followed by a parameter list in parentheses.

Because they are the name of a BASIC file in memory, extension function names must begin with a letter. If it is the first item in a cell entry, the function name must be preceded by the “at” sign (`@`), or the plus sign (`+`), so that VisiCalc can distinguish it from a label. Remember also that the parameter list can contain a range of cell references as above, or a list consisting of ranges and values.

When a cell entry containing an extension function is recalculated, and the BASIC program containing the function is not in memory, the value `NA` will be returned to the cell to indicate that the function is not available.

To better illustrate how the extension function `IRR` works, we will create a new worksheet, called `RETURN`. To begin, enter VisiCalc and create the worksheet:

### Input/Result

```
vc [RTN]
```

Activates the VisiCalc program.

```
VisiCalc (R) - copyright HP 1983
Enter worksheet name: BUDGET
```

VisiCalc prompts for the worksheet file name.

```
[CLR] return [RTN]
```

Enters the new worksheet name.

```
A1:
```

VisiCalc displays the coordinates of the first cell of the worksheet.

Now enter a series of cash flows in cells A1 through A7. To speed up the data entry process, set VisiCalc to Manual Recalculation mode before you start.

/
G
R
M

The **/GRM** command sets VisiCalc to Manual Recalculation mode.

A1:

-10000
RTN
↓

Enters the first cash flow.

1500
RTN
↓

Enters the second cash flow.

2000
RTN
↓

Third cash flow.

2500
RTN
↓

And so on.

3000
RTN
↓

3500
RTN
↓

4000
RTN

After you enter these cash flows, the worksheet should contain the following entries:

RETURN	
	A
-----+-----	
1:	-10000
2:	1500
3:	2000
4:	2500
5:	3000
6:	3500
7:	4000

Now move to cell A8 and enter a call to the IRR extension function.

### Input/Result

↓ @irr(a1...a7)

Before you press **RTN**, VisiCalc displays the function call.

A8(V): @irr(a1...a7)■

**RTN**

Whenever this formula is calculated and the IRR function is called, the HP-75 PRGM annunciator will turn on to indicate that an extension-function program is running.

Working PRGM

A8└ .13871172916

When the extension function finishes processing, the PRGM annunciator turns off and VisiCalc displays the result computed by the function.

The value displayed in cell A8 is the decimal value of the IRR that results from the series of cash flows in cells A1 through A7—it represents a 13.87% IRR.

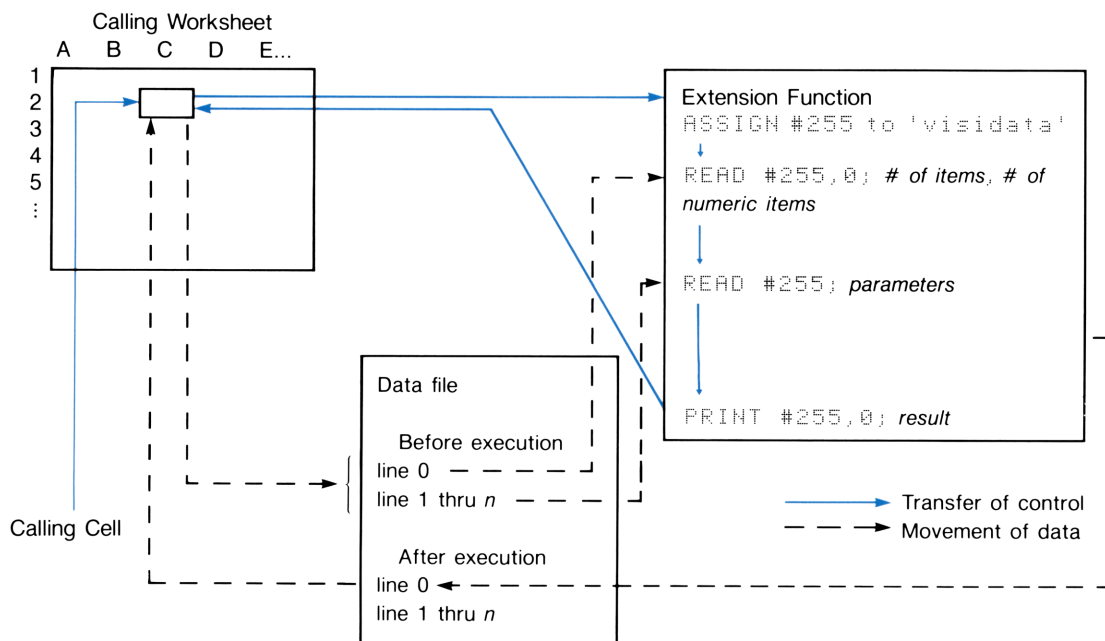
## Passing Values To and From Extension Functions

Like VisiCalc function calls, extension function calls use parameter lists to pass numeric values to the function for processing. The parameters in an extension function parameter list are passed between the worksheet and the function through an HP-75 *data file* called VISIDATA. In this section we will describe how this occurs.

**Note:** If you have not studied part III of the *HP-75 Owner's Manual*, "Programming the HP-75," you may have difficulty with some of the concepts and information presented below. You should be familiar with HP-75 BASIC programming before reading the rest of this section.



Whenever VisiCalc encounters a call to an extension function during worksheet recalculation, it creates an empty data file named `VISIDATA` and transfers the data items in the parameter list to this file before calling the function. The extension function must, in turn, read these parameters from the data file, process them to produce a result, then place that result back in the data file. When the function finishes executing, VisiCalc reads the result from `VISIDATA`, then places it back in the calling cell. This process occurs automatically whenever the worksheet is recalculated, and will work correctly as long as you have entered the function call in the proper form, and the extension function contains the instructions to correctly access the data file. The illustration below shows how this process works:



After VisiCalc finishes writing the data in the parameter list to `VISIDATA`, the data file will have the following general form:

```
0 DATA number of items , number of numeric items
1 DATA data item 1
2 DATA data item 2
3 DATA data item 3
.
.
.
n DATA data item n
```



Line 0 of VISIDATA contains two values; lines 1 through  $n$  contain one value each. The two values in line 0 will be the same if every element of the parameter list is a numeric data item. If there are any nonnumeric items in the parameter list, such as labels or empty cells, then the second value in line 0 will be smaller than the first. In lines 1 through  $n$ , nonnumeric items appear as zeros.

The reason that a maximum of 9,999 parameters can be passed to an extension function is now clear. VisiCalc puts one parameter in each line of VISIDATA starting at line 1, and the largest line number in a data file is 9999, so the number of parameters is determined by the maximum size of the data file. In practice, however, the number of parameters that can be passed is limited by available memory.

The following example will help clarify how VisiCalc passes values to VISIDATA.

**Example:**

If the following range of cells were passed to VISIDATA:

	A
1	-10000
2	3000
3	4000
4	5000
5	-----
6	6000
7	7000
8	0
9	9000
10	-----

By the following function call:

```
IRR(A1...A10)
```

Then `VISIDATA` would contain the following `DATA` statements:

0 DATA 10,8

The second item in this line is the number of items passed minus the number of nonnumeric items passed.

1 DATA -10000

2 DATA 3000

3 DATA 4000

4 DATA 5000

5 DATA 0

Nonnumeric item, passed as zero.

6 DATA 6000

7 DATA 7000

8 DATA 0

A numeric item with value zero.

9 DATA 9000

10 DATA 0

Nonnumeric item, passed as zero.

Notice that, once the parameters are passed to `VISIDATA`, there is no way to tell whether a zero data item is a numeric parameter whose value happens to be zero (line 8, above) or a nonnumeric item from the parameter list (lines 5 and 10, above). In some types of extension functions, this ambiguity causes no problem since you *can* tell how many of the zero data items are numeric by comparing the two values in line 0 of the data file. In functions where the *order* of the items in the parameter list is not important—such as `SUM` and `MEAN`—the parameter list above would be perfectly acceptable. In other functions, like the `IRR` function, the order of the parameters *is* important, and parameter lists like the one above would cause problems. In this case the function would have to be able to distinguish such invalid lists from valid ones.

The extension-function program must be designed to correctly read and interpret the data items in `VISIDATA`; it must also be designed to determine if the values passed to it are acceptable data for the calculations that it will perform. More information about how to correctly design your functions will be given under “Creating Your Own Extension Functions”.

After it finishes its computations, the extension function must return the result of those computations to line 0 of `VISIDATA`. This result must overwrite the original data in line zero, because VisiCalc will return the result `ERROR` to the calling cell if it finds the original data still there. In fact, this is how the function informs VisiCalc that an error has occurred in processing the function—by simply not returning a value to `VISIDATA`.

After the function has completed processing and VisiCalc checks for the result, `VISIDATA` will be purged by VisiCalc to conserve user memory. Each time an extension function call occurs in a worksheet recalculation, `VISIDATA` is re-created.

## Creating Your Own Extension Functions

All extension functions share certain design features and characteristics. In this section, we will examine the fundamental design elements of an extension function, and create some simple extension functions that correctly use these elements.

**Note:** Again we suggest that if you have not studied part III of the *HP-75 Owner's Manual*, "Programming the HP-75," you read that part of the owner's manual before continuing with the rest of this section.

### Input/Output Operations

Since every extension function must use the `VISIDATA` file to pass parameters and results, your functions must contain the instructions necessary to open, read from, and write to the data file. To do this you will need to use the `ASSIGN #`, `READ #`, `PRINT #`, and perhaps the `RESTORE #` BASIC statements.

For example, line 1510 of the `IRR` function opens `VISIDATA`:

```
1510 ASSIGN # 255 TO 'visidata'
```

`VISIDATA` was assigned to file number 255 because it is not likely that another active data file assignment would use a file number this big. Any file number in the valid range of 1 to 9999 could be used, of course.

In line 1550 of `IRR` the *random* `READ #` statement was used to input the number of data items and the number of numeric data items in the parameter list. (The values in the parameter list were placed in `VISIDATA` by VisiCalc.)

```
1550 READ # 255,0 ; N,N1 @ IF N#N1 OR N>255 THEN END
```

The two values in line zero are read into the variables `N` and `N1` respectively. The `IF` statement that follows the `READ #` on the same line tests to see if the parameter list is valid. If it isn't, execution of the function is stopped immediately—the `IRR` can't be computed if the parameter list contains nonnumeric data or is too big.

---

\* A complete listing of the `IRR` function is included in appendix C, "The `IRR` Extension Function" (page 178).

In line 1590 of the function the *serial* `READ #` statement was used to store the parameters into an array that was prepared for them by the `DIM` statement in line 1460:

```
1590 FOR J=1 TO N @ READ # 255 ; C(J) @ NEXT J
```

The `FOR` loop causes each line of `VISIDATA` to be read in sequence until as many lines have been read as there are items in the parameter list.

The last line of the function, line 2110, contains the statements that take care of the last required data file operation:

```
2110 PRINT # 255,0 ; I
```

Line 2110 stores the result, `I` (`I` is the variable that contains the computed `IRR`), into line 0 of `VISIDATA` by means of a random `PRINT #` statement. When this occurs, the original values, which were placed into line 0 by VisiCalc, are destroyed.

These four statements perform all of the input and output required by the `IRR` function.

**Note:** Normally a program that uses a data file should close the data file with an `ASSIGN # file number TO *` statement. In extension functions however, you do not need to include a statement to close the association between `VISIDATA` and the file number, because `VISIDATA` is purged by VisiCalc after the extension function finishes executing, and the association is automatically terminated by that action.

Now examine the following function:

```
10 ! FUNCTION: COUNT(list)
20 ASSIGN # 255 TO 'visidata' ! Open the data file.
30 READ # 255,0 ; N,N ! Read the number of numeric parameters.
40 PRINT # 255,0 ; N ! Restore that number back into line 0.
```

This simple function takes advantage of the way VisiCalc passes information to an extension function to return, without any computation, the number of numeric parameters in the list sent to it. As you can see, even this simple function contains all of the essential input/output operations that every extension function must have.

## Trapping Input Errors

One of the characteristics of a well designed extension function is the ability to “trap” input data errors. You’ve already seen an example of such error trapping in the `IF` statement in line 1550 of the `IRR` function. Recall that the purpose of this statement is to test for a nonvalid parameter list.

The nature of the computations performed by an extension function will dictate the types of error trapping that must be done in the function. In general, you will want to test for input data that will:

- Produce an invalid result because the algorithm used won’t handle the data. For example, the `IRR` function may produce more than one answer if there are several negative cash flows separated by one or more positive cash flows, so a data set with this characteristic is trapped as an error.
- Not produce a result at all. For example, the `IRR` routine can’t compute an answer if there are no negative cash flows, so this condition is also trapped.

In the `IRR` routine, lines 1650 through 1850 contain tests for various types of invalid input data that cause the function to terminate, when error conditions are found, without returning a value to line 0 of `VISIDATA`. Most extension functions will need an analogous section of code to test for and trap errors in their input data.

## Putting Together a Model

By combining the elements discussed above with the required characteristics of a function, we can develop a general model that can be used as a guide for creating extension functions. As this model is developed below, it will be used to design an extension function, called `STDEV`, that computes the standard deviation,  $s$ , of a set of data points. To design `STDEV` we need to answer several questions about the function:

**What are the inputs of the function?** We will define the input as a standard VisiCalc parameter list containing any combination of numeric and nonnumeric items. Further, we will not place a limit on the size of the parameter list. (Remember that VisiCalc limits the number of parameters in a list sent to an extension function to 9,999.) We can do this because we can compute the standard deviation without storing all of the parameters in separate variables and thus do not need to reserve space for a large number of variables.



**What kind of input errors must be trapped?** Since the order of the parameters is unimportant, it's not necessary to terminate execution if one or more nonnumeric parameters are passed to the function. The equation that we'll use to compute the standard deviation requires the division of an expression by the number of numeric parameters, so if no numeric parameters are passed, a divide-by-zero error will occur. However, if the HP-75 is set to `DEFAULT OFF`, when a mathematical error occurs the function will terminate and control will be returned to VisiCalc, so these types of errors also do not need to be trapped.

**How will the result be computed?** There are several formulas for computing the standard deviation. The one we will use is:

$$s_x = \frac{n\sum x^2 - (\sum x)^2}{n^2 - n}$$

Using this list of specifications and the model we will develop below, you can design the function.

**Note:** The items in the model below are listed in an order that represents their recommended order in the program.

1. Initialize and define the variables. The best place to define and initialize any variables used in the function is at the beginning of the program. For the `STDEV` function six simple numeric variables are needed:

- X     A temporary variable to hold the value of the last parameter read in.
- X1    The running sum of the parameters read so far.
- X2    The running sum of the squares of the parameters read so far.
- N     The number of numeric parameters passed.
- S     The standard deviation.
- I     A loop counter variable.

Variables `X1` and `X2` must be initialized to value 0 before they can be used. The other variables don't need to be initialized.

The first part of the function looks like this:

```
1000 ! FUNCTION: STDEV(list)
1100 DEFAULT OFF ! Causes math errors to terminate execution.
1200 X1,X2=0 ! Initialize X1 and X2.
```

2. Open the data file. `VISIDATA` must be assigned a file number so that the function can access it:

```
1300 ASSIGN # 255 TO 'visidata'
```

Remember that you can use any valid file number, but be careful to avoid conflicts with other data file assignments that might be active.

3. Read line 0 of the data file to get the number of parameters and the number of numeric parameters. In this function, since nonnumeric parameters are passed as zeros, the only important value is the number of numeric parameters—nonnumeric parameters will have no effect on the computation:

```
1400 READ # 255,0 ; N,N
```

The variable `N` was used to read both values, since it's the *second* of the two values that's of interest. When this statement is executed, the number of parameters will be stored in `N` first, then the number of numeric parameters will replace the first value stored into `N`.

4. Read in and store the parameters. In this function we will read the parameters and immediately compute some intermediate values that will be used in the final computation. This means that it won't be necessary to provide storage for all of the parameters—just the current one. In the `IRR` function it *was* necessary to store all of the parameters before computation could begin, so the variables used by that program took up much more memory. Here is the part of `STDEV` that reads the parameters and does the intermediate computations:

```
1500 FOR I = 1 TO N @ READ # 255 ; X ! Read a parameter.
1600 X1=X1+X @ X2=X2+X*X ! Add the parameter to X1 and its square to X2.
1700 NEXT I ! Get another parameter.
```

5. Test for errors. In this function there aren't any errors to test for, so there aren't any error trapping instructions in the function. Remember that in the `IRR` function, lines 1650 through 1850 contain the error checking routine.
6. Compute the function. Here is where the result that will be returned to VisiCalc is computed. In the `STDEV` function the computation involves only a relatively simple formula, but in others it may be more complex—as in the `IRR` function, for example. Here is the computation section of the `STDEV` function:

```
1800 S = SQR((N*X2-(X1*X1))/(N*N-N))
```

7. Return the result to VisiCalc:

```
1900 PRINT # 255,0 ; S
```

This model contains the elements that will be common to virtually all extension functions. As you can see from the `STDEV` function and the `COUNT` function, some functions will not contain all of the listed elements. Here is the `STDEV` function as it might look after the program is compressed into a smaller space to save memory:

```
1000 ! FUNCTION: STDEV(list)
1100 DEFAULT OFF @ X1,X2=0 @ ASSIGN # 255 TO 'visidata'
1400 READ # 255,0 ; N,N
1500 FOR I = 1 TO N @ READ # 255 ; X @ X1=X1+X @ X2=X2+X*X @ NEXT I
1800 S = SQR((N*X2-X1*X1)/(N*N-N)) @ PRINT # 255,0 ; S
```

The catalog entry for this BASIC file shows that the file occupies 214 bytes of memory—very compact indeed! When you create your own extension functions, you might want to maintain two versions of the function: one that is compressed as in the listing above, and a second that is fully documented. The reason for both versions is simple. You want your extension functions to use as little memory as possible, so as to leave plenty of room for worksheets and other files in memory. At the same time a fully documented program is much easier to modify or update at some time in the future. The easiest way to reconcile these somewhat incompatible requirements is to maintain two complete versions—one for execution and one for documentaton. The listing of the `IRR` function in appendix C illustrates this idea. It is the fully documented version of the program that is actually recorded on the magnetic card you received.

## Avoiding and Correcting Design Errors

Since an extension function is simply a special kind of BASIC program, you should use all of the good programming practices that you learned from part III of your *HP-75 Owner's Manual* when designing your own functions. Listed below are several additional programming points to consider when designing extension functions:

- In an extension function, do not execute `RUN`, `CLEAR VARS`, or any file manipulation command that causes the extension function to be deallocated or removed from memory.\* These commands cause all existing variable assignments in memory to be cleared—including those used by VisiCalc. If any of these commands are executed by an extension function, both the extension function and VisiCalc will be halted and the HP-75 will display error 202—in extension fn. In addition, the worksheet may be inconsistent when you restart VisiCalc. Before recalculating the worksheet again, remove the offending commands from your extension function to prevent a reoccurrence of the error condition.

\* Refer to “Using File Commands With Allocated Programs,” in section 11 of the *HP-75 Owner's Manual*, for information about which commands deallocate programs.



- All parameters passed between VisiCalc and an extension function are *real* values. Do not include `INTEGER` or `SHORT` declarations in extension functions, since a passed value that is too large to fit into the limits of `SHORT` and `INTEGER` variables will cause an error during execution of the function.
- If you place the `BYE` command in an extension function, be sure that the VisiCalc module is installed in the HP-75 before turning on the computer after the `BYE` is executed; failure to do so will require a system reset to recover.
- If you create an extension function with a file name that is the same as the name of a VisiCalc built-in function or even an HP-75 function, your function—and not the built-in functions—will always get executed by VisiCalc. You can use this feature to your advantage to create your own version of a built-in or HP-75 function.
- Any condition that causes an extension function to halt before the normal end of execution, will terminate the extension function and cause VisiCalc to return `ERROR` to the calling cell. The conditions that cause this to occur are:
  - Pressing the `ATTN` key during execution of the extension function.
  - Executing a `STOP` instruction in the extension function.
  - Any error that halts program execution (that is, any error that would normally cause an HP-75 error message to be displayed).

It is difficult to test and debug extension functions by calling them from a worksheet. The normal error checking and diagnostic tools that are available in HP-75 BASIC do not work well from a worksheet because all conditions that halt an extension function (`ATTN`, `STOP`, and errors) simply cause an `ERROR` value to be returned to the calling cell. Because of this, the easiest way to test extension functions is to create your own `VISIDATA` file, store some test parameters directly into the data file in the format illustrated in “Passing Values To and From Extension Functions” (page 155), then execute the extension function from the keyboard and observe the effect on the `VISIDATA` file.

When you execute your function, it will read values from and store values to the `VISIDATA` file just as if VisiCalc had created the data file and placed the values there. By examining line 0 of the `VISIDATA` file after the extension function has completed, you can see if the function correctly computed the desired result and stored it in the data file.

You can also incorporate various debugging instructions (like `TRACE VARS`, `TRACE ALL`, etc.) into the function to trace the operation of the function while it is running. After your testing and debugging is completed, you can remove the debugging instructions so the function can be used with VisiCalc.

## LEX Files in RAM or ROM

VisiCalc also allows formulas to include keywords from LEX files that reside in user memory or in plug-in modules. The keywords that can be used are those which take either no parameters or only simple numeric parameters (any number of parameters are allowed), and return a single numeric result. Keywords that require string or numeric array parameters, or that return a string, an array, or nothing at all, are not allowed. LEX function parameter lists, like those of HP-75 functions accessible to VisiCalc, must consist of single parameters separated by commas—ranges are not allowed.

When a formula containing a LEX function is displayed, and the LEX file has been purged or the plug-in module containing the function has been removed from the computer, the LEX function name will be replaced—in the display—with: `LEX n, m` (where *n* is the ROM identification number of the LEX file, and *m* is the function number of the function).<sup>\*</sup> When the formula containing the missing LEX function is recalculated, the value `NA` will be returned to the cell to indicate that the function is not available.

Recall from “Displaying a Formula or a Result” (page 62), that under certain conditions a formula may become too long to be displayed, and that this can cause problems if you attempt to edit the formula. A similar situation will occur if a formula containing a missing LEX function is displayed—even if the formula is still less than 91 characters long. To edit the formula, you must reinstall the LEX function or remove the function reference from the formula. Also, in the `Both` option of the Replicate command, if a missing LEX function message causes the formula to be longer than 91 characters, VisiCalc will replicate the formula as if you had selected the `No change` option. The only way to specify the `Both` option with such a formula, is to reinstall the LEX function in the computer.

---

<sup>\*</sup> The documentation for the LEX file you are using will give the ROM identification number and the name of the function associated with displayed function number.

## Appendixes and Indexes

The **appendixes** and **indexes** provide additional reference information.

- **Appendix A** (“Owner’s Information”) discusses care and service of the VisiCalc module.
- **Appendix B** (“VisiCalc Functions”) lists the functions available for use with VisiCalc.
- **Appendix C** (“The IRR Extension Function”) provides a commented listing of the IRR extension function.
- **Appendix D** (“Changing to Foreign Language Prompts”) describes how to change the VisiCalc prompts and messages from English to another language.
- **Appendix E** (“Setting the Video Display Parameters”) describes how to change the video display parameters to use VisiCalc with other display devices.
- **Appendix F** (“Glossary”) is a glossary of terms used in this manual.
- The **“Command and Operation Summary and Index”** provides a comprehensive summary of the operations, commands, and functions of VisiCalc. This summary is more extensive than the *VisiCalc Quick Reference Guide*, and includes the page number of the principal coverage of the topic in the manual. It is designed to be your principal reference tool after you have learned to use VisiCalc.
- The **“Subject Index”** provides a complete alphabetical index of the topics covered by this manual. It will assist you in finding information in the manual.

## Appendix A

# Owner's Information

## Contents

Maintenance .....	170
Limited One-Year Warranty .....	170
What We Will Do .....	170
What Is Not Covered .....	170
Warranty for Consumer Transactions in the United Kingdom .....	171
Obligation to Make Changes .....	171
Warranty Information .....	171
Service .....	172
Service Centers .....	172
Obtaining Repair Service in the United States .....	172
Obtaining Repair Service in Europe .....	172
International Service Information .....	173
Service Repair Charge .....	173
Service Warranty .....	173
Shipping Instructions .....	173
Further Information .....	174
Technical Assistance .....	174
Dealer and Product Information .....	174

## Maintenance

The HP-75 VisiCalc module does not require maintenance. However, there are several precautions, listed below, that you should observe.

### CAUTIONS

- Do not place fingers, tools, or other objects into the plug-in ports. Damage to plug-in module contacts and the computer's internal circuitry may result.
- Turn off the computer (press **[SHIFT]** **[ATTN]**) before installing or removing a plug-in module.
- If a module jams when inserted into a port, it may be upside down. Attempting to force it further may result in damage to the computer or the module.
- Handle the plug-in modules very carefully while they are out of the computer. Do not insert any objects in the module connector socket. Always keep a blank module in the computer's port when a module is not installed. Failure to observe these cautions may result in damage to the module or the computer.

## Limited One-Year Warranty

### What We Will Do

VisiCalc is warranted by Hewlett-Packard against defects in materials and workmanship affecting electronic and mechanical performance, but not software content, for one year from the date of original purchase. If you sell your unit or give it as a gift, the warranty is transferred to the new owner and remains in effect for the original one-year period. During the warranty period, we will repair or, at our option, replace at no charge a product that proves to be defective, provided you return the product, shipping prepaid, to a Hewlett-Packard service center.

### What Is Not Covered

This warranty does not apply if the product has been damaged by accident or misuse or as the result of service or modification by other than an authorized Hewlett-Packard service center.

No other express warranty is given. The repair or replacement of a product is your exclusive remedy. **ANY OTHER IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS IS LIMITED TO THE ONE-YEAR DURATION OF THIS WRITTEN WARRANTY.** Some states, provinces, or countries do not allow limitations on how long an implied warranty lasts, so the above limitation may not apply to you. **IN NO EVENT SHALL HEWLETT-PACKARD COMPANY BE LIABLE FOR CONSEQUENTIAL DAMAGES.** Some states, provinces, or countries do not allow the exclusion or limitation of incidental or consequential damages, so the above limitation or exclusion may not apply to you.

This warranty gives you specific legal rights, and you may also have other rights that vary from state to state, province to province, or country to country.

## Warranty for Consumer Transactions in the United Kingdom

This warranty shall not apply to consumer transactions and shall not affect the statutory rights of a consumer. In relation to such transactions, the rights and obligations of Seller and Buyer shall be determined by statute.

## Obligation to Make Changes

Products are sold on the basis of specifications applicable at the time of manufacture. Hewlett-Packard shall have no obligation to modify or update products once sold.

## Warranty Information

If you have any questions concerning this warranty, please contact an authorized Hewlett-Packard dealer or a Hewlett-Packard sales and service office. Should you be unable to contact them, please contact:

- In the United States:

Hewlett-Packard Company  
Portable Computer Division  
1000 N.E. Circle Blvd.  
Corvallis, OR 97330  
Telephone: (503) 758-1010

Toll-Free Number: (800) 547-3400 (except in Oregon, Hawaii, and Alaska)

- In Europe:

Hewlett-Packard S.A.  
150, route du Nant-d'Avril  
P.O. Box CH-1217 Meyrin 2  
Geneva  
Switzerland  
Telephone: (022) 83 81 11

**Note:** Do *not* send units to this address for repair.

- In other countries:

Hewlett-Packard Intercontinental  
3495 Deer Creek Rd.  
Palo Alto, California 94304  
U.S.A.  
Telephone: (415) 857-1501

**Note:** Do *not* send units to this address for repair.



# Service

## Service Centers

Hewlett-Packard maintains service centers in most major countries throughout the world. You may have your unit repaired at a Hewlett-Packard service center any time it needs service, whether the unit is under warranty or not. There is a charge for repairs after the one-year warranty period.

Hewlett-Packard products are normally repaired and reshipped within five (5) working days of receipt at any service center. This is an average time and could vary depending upon the time of year and the work load at the service center. The total time you are without your unit will depend largely on the shipping time.

## Obtaining Repair Service in the United States

The Hewlett-Packard United States Service Center for battery-powered computational products is located in Corvallis, Oregon:

Hewlett-Packard Company  
Service Department  
P.O. Box 999  
Corvallis, Oregon 97339, U.S.A.  
or  
1030 N.E. Circle Blvd.  
Corvallis, Oregon 97330, U.S.A.  
Telephone: (503) 757-2000

## Obtaining Repair Service in Europe

Service centers are maintained at the following locations. For countries not listed, contact the dealer where you purchased your unit.

### AUSTRIA

HEWLETT-PACKARD Ges.m.b.H.  
Kleinrechner-Service  
Wagramerstrasse-Lieblgasse 1  
A-1220 Wien (Vienna)  
Telephone: (0222) 23 65 11

### BELGIUM

HEWLETT-PACKARD BELGIUM SA/NV  
Woluwedal 100  
B-1200 Brussels  
Telephone: (02) 762 32 00

### DENMARK

HEWLETT-PACKARD A/S  
Datavej 52  
DK-3460 Birkerød (Copenhagen)  
Telephone: (02) 81 66 40

### EASTERN EUROPE

Refer to the address listed under Austria.

### FINLAND

HEWLETT-PACKARD OY  
Revontulentie 7  
SF-02100 Espoo 10 (Helsinki)  
Telephone: (90) 455 02 11

### FRANCE

HEWLETT-PACKARD FRANCE  
Division Informatique Personnelle  
S.A.V. Calculateurs de Poche  
F-91947 Les Ulis Cedex  
Telephone: (6) 907 78 25

### GERMANY

HEWLETT-PACKARD GmbH  
Kleinrechner-Service  
Vertriebszentrale  
Berner Strasse 117  
Postfach 560 140  
D-6000 Frankfurt 56  
Telephone: (611) 50041

### ITALY

HEWLETT-PACKARD ITALIANA S.P.A.  
Casella postale 3645 (Milano)  
Via G. Di Vittorio, 9  
I-20063 Cernusco Sul Naviglio (Milan)  
Telephone: (2) 90 36 91

**NETHERLANDS**

HEWLETT-PACKARD NEDERLAND B.V.  
 Van Heuven Goedhartlaan 121  
 NL-1181 KK Amstelveen (Amsterdam)  
 P.O. Box 667  
 Telephone: (020) 472021

**SPAIN**

HEWLETT-PACKARD ESPANOLA S.A.  
 Calle Jerez 3  
 E-Madrid 16  
 Telephone: (1) 458 2600

**SWITZERLAND**

HEWLETT-PACKARD (SCHWEIZ) AG  
 Kleinrechner-Service  
 Allmend 2  
 CH-8967 Widen  
 Telephone: (057) 31 21 11

**NORWAY**

HEWLETT-PACKARD NORGE A/S  
 P.O. Box 34  
 Oesterndalen 18  
 N-1345 Oesteraas (Oslo)  
 Telephone: (2) 17 11 80

**SWEDEN**

HEWLETT-PACKARD SVERIGE AB  
 Skalholtsgatan 9, Kista  
 Box 19  
 S-163 93 Spanga (Stockholm)  
 Telephone: (08) 750 20 00

**UNITED KINGDOM**

HEWLETT-PACKARD Ltd  
 King Street Lane  
 GB-Winnersh, Wokingham  
 Berkshire RG11 5AR  
 Telephone: (0734) 784 774

## International Service Information

Not all Hewlett-Packard service centers offer service for all models of HP products. However, if you bought your product from an authorized Hewlett-Packard dealer, you can be sure that service is available in the country where you bought it.

If you happen to be outside of the country where you bought your unit, you can contact the local Hewlett-Packard service center to see if service is available for it. If service is unavailable, please ship the unit to the address listed above under "Obtaining Repair Service in the United States." A list of service centers for other countries can be obtained by writing to that address.

All shipping, reimportation arrangements, and customs costs are your responsibility.

## Service Repair Charge

There is a standard repair charge for out-of-warranty repairs. The repair charges include all labor and materials. In the United States, the full charge is subject to the customer's local sales tax. In European countries, the full charge is subject to Value Added Tax (VAT) and similar taxes wherever applicable. All such taxes will appear as separate items on invoiced amounts.

Products damaged by accident or misuse are not covered by the fixed repair charges. In these situations, repair charges will be individually determined based on time and materials.

## Service Warranty

Any out-of-warranty repairs are warranted against defects in materials and workmanship for a period of 90 days from date of service.

## Shipping Instructions

Should your unit require service, return it with the following items:

- A completed Service Card, including a description of the problem.
- A sales receipt or other proof of purchase date if the one-year warranty has not expired.

The product, the Service Card, a brief description of the problem, and (if required) the proof of purchase date should be packaged in adequate protective packaging to prevent in-transit damage. Such damage is not covered by the one-year limited warranty; Hewlett-Packard suggests that you insure the shipment to the service center. The packaged unit should be shipped to the nearest Hewlett-Packard designated collection point or service center. Contact your dealer for assistance. (If you are not in the country where you originally purchased the unit, refer to "International Service Information," above.)

Whether the unit is under warranty or not, it is your responsibility to pay shipping charges for delivery to the Hewlett-Packard service center.

After warranty repairs are completed, the service center returns the unit with postage prepaid. On out-of-warranty repairs in the United States and some other countries, the unit is returned C.O.D. (covering shipping costs and the service charge).

## Further Information

Service contracts are not available. Circuitry and designs are proprietary to Hewlett-Packard, and service manuals are not available to customers. Should other problems or questions arise regarding repairs, please call your nearest Hewlett-Packard service center.

## Technical Assistance

The keystroke procedures and program material in this manual are supplied with the assumption that the user has a working knowledge of the concepts and terminology used. Hewlett-Packard's technical support is limited to explanations of operating procedures used in the manual and verification of results obtained in the examples. If you have technical problems involving this manual, first consult your *HP-75 Owner's Manual*. Should you need further assistance, you may write to:

Hewlett-Packard  
Portable Computer Division  
Customer Support  
1000 N.E. Circle Blvd.  
Corvallis, OR 97330

## Dealer and Product Information

For U.S.A. dealer locations, product information, and prices, please call (800) 547-3400. In Oregon, Alaska, and Hawaii, call (503) 758-1010.

## Appendix B

# VisiCalc® Functions

This is a list of the functions available for use with VisiCalc and the HP-75. In the following table, functions that are unique to VisiCalc are marked “VC” and are described in detail elsewhere in this manual. HP-75 functions that are usable with VisiCalc are described in detail in the *HP-75 Owner’s Manual*.

In the following list,  $x$  and  $y$  specify any two numeric expressions, and you can use values or cell references as parameters. The asterisks following the explanation for trigonometric functions refer to the footnote at the bottom of page 177.

Function	Explanation
ABS( $x$ )	Absolute value of $x$ .
ACOS( $x$ )	Arccosine of $x$ , in Quadrant I or II.*
ANGLE( $x, y$ )	Arctangent of $y/x$ , in proper quadrant. That is, returns the angle $\theta$ formed between the $x$ -axis and the point $(x,y)$ , such that $-\pi < \theta \leq \pi$ .
ASIN( $x$ )	Arcsine of $x$ , in Quadrant I or V.*
ATN( $x$ )	Arctangent of $x$ , in Quadrant I or IV.*
AVERAGE( <i>list</i> )	VC Computes the arithmetic mean or average of the numeric parameters in the list. Ignores non-numeric parameters.
CEIL( $x$ )	Smallest integer $\geq x$ .
COS( $x$ )	Cosine of $x$ .*
COT( $x$ )	Cotangent of $x$ .*
CSC( $x$ )	Cosecant of $x$ .*
DATE	The date in <i>yyddd</i> format based on the clock setting.
DEG( $x$ )	Radian-to-degree conversion of $x$ .
EPS	Smallest machine number ( $1.E-499$ ).
ERRL	The line number of the most recent error or warning.
ERRN	The identification number of the most recent error or warning.
ERROR	VC Results in an “Error” value that makes all expressions using the value display ERROR in the cell display.
EXP( $x$ )	$e^x$ .

Function	Explanation
FLOOR( <i>x</i> )	Same as INT( <i>x</i> ).
FP( <i>x</i> )	Fractional part of <i>x</i> .
INF	Largest machine number (9.999999999999E499).
INT( <i>x</i> )	Largest integer $\leq x$ .
IP( <i>x</i> )	Integer part of <i>x</i> .
LOG( <i>x</i> )	Natural logarithm of <i>x</i> , $x > 0$ .
LOG10( <i>x</i> )	Log to the base 10 of <i>x</i> , $x > 0$ .
MAX( <i>x</i> , <i>y</i> )	If $x > y$ then <i>x</i> , else <i>y</i> .
MAXL( <i>list</i> )	VC Returns the maximum value in the <i>list</i> .
MEAN( <i>list</i> )	VC Computes the arithmetic mean of the values in the <i>list</i> . MEAN is another name for the AVERAGE function.
MEM	The number of bytes of available memory.
MIN( <i>x</i> , <i>y</i> )	If $x < y$ then <i>x</i> , else <i>y</i> .
MINL( <i>list</i> )	VC Returns the minimum value in the <i>list</i> .
MOD( <i>x</i> , <i>y</i> )	<i>x</i> modulo <i>y</i> : $x - y * \text{INT}(x/y)$ .
NA	VC Returns a “Not Available” value that makes all expressions using the value display NA. Allows the effect of a particular value to be traced through the worksheet.
PI	3.14159265359.
RAD( <i>x</i> )	Degree-to-radian conversion of <i>x</i> .
RES	The last numeric result to be displayed or printed.
RMD( <i>x</i> , <i>y</i> )	Remainder of $x/y$ : $x - y * \text{IP}(x/y)$ .
RND	Next number, <i>r</i> , in sequence of pseudo-random numbers, $0 \leq r < 1$ .

Function	Explanation
SEC( <i>x</i> )	Secant of <i>x</i> .*
SGN( <i>x</i> )	The sign of <i>x</i> : -1 if $x < 0$ , 0 if $x = 0$ , 1 if $x > 0$ .
SIN( <i>x</i> )	Sine of <i>x</i> .*
SQR( <i>x</i> )	Positive square root of <i>x</i> .
SUM( <i>list</i> )	VC Computes the sum of the values in the <i>list</i> .
TAN( <i>x</i> )	Tangent of <i>x</i> .*
TIME	The number of seconds since midnight of the current day.

---

\* The trigonometric functions operate in the current trigonometric setting of the HP-75 (OPTION ANGLE RADIANS or OPTION ANGLE DEGREES).

## Appendix C

# The IRR Extension Function

```

1000 ! *****
1010 ! *
1020 ! * FUNCTION: IRR(list).
1030 ! * Computes a unique positive IRR of an ungrouped series of cash
1040 ! * flows.
1050 ! *
1060 ! * CREATED ON: 11/30/82 REVISIED ON: 4/26/83
1070 ! *
1080 ! * INPUTS:
1090 ! * Standard VisiCalc parameter list read from file VISIDATA. All
1100 ! * elements of the list must be numeric. The number of elements
1110 ! * must be <= 255. Cash flows with value 0 are valid.
1120 ! *
1130 ! * OUTPUTS:
1140 ! * IF data is valid and the IRR is computable, THEN returns the
1150 ! * decimal value of the IRR to line 0 of VISIDATA, ELSE returns
1160 ! * nothing to VISIDATA and hence ERROR to VisiCalc.
1170 ! *
1180 ! * VARIABLES:
1190 ! * X = Max iteration count.
1200 ! * E = Error limit.
1210 ! * I1 = Initial guess of IRR.
1220 ! * C(1..255) = Array of cash flows (from VISIDATA).
1230 ! * N = # of items in input list (from VISIDATA).
1240 ! * N1 = # of numeric entries in input list (from VISIDATA).
1250 ! * J, K = Loop counters.
1260 ! * S1, S2 = Temporary sums of CFs used in error trap routine.
1270 ! * W, G, G1 = Temporary intermediate values used in IRR.
1280 ! * P1, P2, P3,
1290 ! * P4, A, B = Temporary variables used in NPV routine.
1300 ! * I = IRR (result returned to VisiCalc).
1310 ! *
1320 ! * ERROR CHECKS: Routine tests for:
1330 ! * (1) Invalid (nonnumeric) data in input.
1340 ! * (2) All CFs = 0.
1350 ! * (3) Zero or more than one root of the IRR equation.
1360 ! * Errors found cause the routine to return nothing to VISIDATA,
1370 ! * which causes VisiCalc to place the value ERROR in the calling
1380 ! * cell.
1390 ! *
1400 ! *****

```



```

1410 ! *
1420 ! * BEGIN Program IRR.
1430 ! *   Initialize variables.
1440 ! *
1450 OPTION BASE 1 @ DEFAULT OFF
1460 DIM C(255)
1470 X=25 @ E=1*10^-6 @ I1=PI/100
1480 ! *
1490 ! * Open data file 'VISIDATA'.
1500 ! *
1510 ASSIGN # 255 TO 'visidata'
1520 ! *
1530 ! * Read # entries and # numeric entries, then test for invalid data.
1540 ! *
1550 READ # 255,0 ; N,N1 @ IF N#N1 OR N>255 THEN END
1560 ! *
1570 ! * Read data set.
1580 ! *
1590 FOR J=1 TO N @ READ # 255 ; C(J) @ NEXT J
1600 ! *
1610 ! * BEGIN trap errors routine.
1620 ! *
1630 ! *   Test for all CF = 0. Exit if true.
1640 ! *
1650 S1=0
1660 FOR J=1 TO N @ S1=S1+C(J) @ IF S1 THEN 1710
1670 NEXT J @ END
1680 ! *
1690 ! *   Test for no roots to the IRR equation. Exit if true.
1700 ! *
1710 S2=S1
1720 FOR J=J+1 TO N @ S2=S2+C(J)
1730 IF NOT S2 THEN 1750
1740 IF SGN(S1)*SGN(S2) THEN 1800
1750 NEXT J
1760 IF S2=0 THEN I=0 @ GOTO 2110 ELSE END
1770 ! *
1780 ! *   Test for more than 1 root to the IRR equation. Exit if true.
1790 ! *
1800 S1=S2
1810 FOR J=J+1 TO N-1 @ S1=S1+C(J)
1820 IF S1=0 THEN 1840
1830 IF SGN(S2)*SGN(S1) THEN END
1840 NEXT J
1850 IF S1=0 THEN I=0 @ GOTO 2110
1860 ! * END trap errors routine.
1870 ! *
1880 ! * BEGIN compute IRR routine and initialize.
1890 W=1+I1 @ GOSUB 2000 @ K=1
1900 ! *
1910 ! * BEGIN IRR iteration loop.
1920 IF ABS(LOG(P1/(-P2)))<E THEN I=I1 @ GOTO 2110

```

```

1930 W=W*((-P2)/P1)^(1/(W*(P3/P1-P4/P2)))
1940 I1=W-1 @ GOSUB 2000
1950 K=K+1 @ IF K>=X THEN END ELSE GOTO 1920
1960 ! * END IRR iteration loop.
1970 ! * END IRR routine.
1980 ! *
1990 ! * BEGIN npv routine
2000 P1,P2,P3,P4=0
2010 FOR J=1 TO N
2020 IF C(J)<0 THEN A=0 @ B=C(J) ELSE A=C(J) @ B=0
2030 G=W^(-J+1) @ G1=W^(-J)*(1-J)
2040 P1=P1+A*G @ P2=P2+B*G @ P3=P3+A*G1 @ P4=P4+B*G1
2050 NEXT J
2060 I=P1+P2 @ RETURN
2070 ! * END NPV routine
2080 ! *
2090 ! * Return result
2100 ! *
2110 PRINT # 255,0 ; I
2120 ! * END of program.

```

# Changing to Foreign Language Prompts

## Contents

Introduction .....	181
Prompt Substitution .....	182
The Set of Message Elements .....	182
Translation Considerations .....	185
An Example VISIMSGS File .....	185

## Introduction

VisiCalc has been designed so that the prompts and messages displayed by VisiCalc can easily be changed. The principal reason for this capability is to provide for the translation of all prompts and expected keystroke responses into other languages. For example, the VisiCalc prompt:

Delete row 1: Yes No

prompts the user to confirm that row 1 is to be deleted. The **Y** key confirms the deletion, and the **N** key cancels it. Any other keystroke (except for **ATTN** and **SHIFT** **ATTN**) is ignored, and VisiCalc waits for either the **Y** or **N** key.

For a French speaking user, the above prompt might translate to:

Supprime rang 1: Oui Non

with VisiCalc expecting an **O** to confirm deletion, or an **N** to cancel deletion, ignoring all other keystrokes (except, of course, for **ATTN** and **SHIFT** **ATTN**). Such translated prompts are possible because of the way VisiCalc stores and accesses the messages and prompts it uses.

## Prompt Substitution

The prompts and messages displayed by VisiCalc are composed of message elements stored in the VisiCalc module (ROM). VisiCalc constructs prompts and messages for display by selecting the correct message elements from among those stored and placing them in the display. Before looking in the module for message elements however, VisiCalc checks user memory (RAM) for the existence of an HP-75 text file called `VISIMSGS`. If `VISIMSGS` exists, VisiCalc looks in that file for the necessary message elements. Only if the appropriate message elements are not found in `VISIMSGS` does VisiCalc use the messages stored in ROM.

New messages and prompts can be created by placing the proper entries into `VISIMSGS`, which—like all text files—consists of lines of text identified by line numbers. Each line of `VISIMSGS` will contain one message element corresponding to one of those used by VisiCalc. VisiCalc searches `VISIMSGS` by line number and, if it finds a message element on the designated line, uses that element. Since VisiCalc checks `VISIMSGS` each time a message or prompt is required, any particular message can be changed without having a complete set of message elements in RAM.

There are two types of message elements used by VisiCalc. One type is a simple text display, like `Working`—no error checking is done on these messages. The other type is a choice element, like `Yes No`, where underlined characters are used to indicate expected keystroke responses. A choice element in `VISIMSGS` must have the same number of underlined characters as its equivalent in ROM. If there are too *many* choices in a choice element, VisiCalc may return an error message when one of these extra choices is selected. If there are too *few* choices, the user will not have access to all of the command options normally available with that command. Also, the first letter of each option of a choice element should start with a different letter—where letters are the same, the first occurrence of the choice will always be executed.

## The Set of Message Elements

The complete set of message elements used by VisiCalc is listed below in a form that represents how they would appear if they were copied into the `VISIMSGS` file. Because of the way they are stored in ROM, they cannot be copied directly into RAM—although you can recreate the message elements by entering them as lines in `VISIMSGS`.

Line Number	Message Element	Use
1	Enter worksheet name	Start-up message.
2	Working	VisiCalc busy message.
3	(V)	} Top level cell type messages.
4	(L)	
5	(/—)	
6	D	} Status line formats.
7	G	
8	I	
9	L	
10	R	
11	\$	}
12	*	
13	C	} Status line recalculation orders.
14	R	
15	A	} Status line recalculation modes.
16	M	
17	V	} Status line cell type messages.
18	L	
19	/—	
20	Go to	Go To message.
21	Command	} Command level messages.
22	DEFGHIMPRVW=	
23	Yes No	Choice message.
24	Delete	} Delete command messages.
25	Col Row	
26	Delete col	
27	Delete row	} Format command messages.
28	Format (local)	
29	D G I L R \$ *	} Global command messages.
30	Global	
31	Col Order Recalc Format	

Line Number	Message Element	Use
32	Col width (global)	Format command messages.
33	Order of recalc	
34	Recalc	
35	Auto Manual	
36	Format (global)	
37	G I L R ± ±	Header command messages.
38	Headers	
39	Col Row Width Suppress	
40	Col header	
41	Row header	
42	Width row headers	Insert command message.
43	Suppress default headers	
44	Insert	
45	Move	
46	Move col	
47	Move row	Move command messages.
48	to	
49	Pr	
50	Worksheet +Headers Formulas	
51	Upper left	
52	Lower right	Print command messages.
53	Printer setup	
54	Printer width	
55	Leaving worksheet	
56	Repl source	
57	Repl target	Exit message.
58	Repl	
59	No change Relative Both	
60	No Rel	
61	Video active	
62	Width of col	Width command message.

## Translation Considerations

Note that some message elements are used to make more than one prompt:

- Message element 23 is used with message elements 26, 27, 43, and 61.
- Message element 25 is used with message elements 24, 44, and 45.
- Message element 48 is used with message elements 46 and 47.

Message elements should be carefully chosen so they combine well with other elements. Care must be taken in translation to see that the words and messages chosen are used consistently. Care should also be taken to choose the appropriate length for message elements. Some elements should be kept short so they will fit properly in the 32 character display window of the HP-75.

When entering choice message elements into VISIMSGS, you must enter underlined characters for the choices. Since many of the HP-75's underlined characters are not available from the keyboard, VisiCalc provides a keystroke sequence similar to `[SHIFT][I/R]` that allows you to enter underlined characters without using the `CHR#` function. If you press `[CTL][I/R]`, the next character key pressed will be entered and displayed as its underlined equivalent. This keystroke sequence is only available when the VisiCalc module is plugged in.

## An Example VISIMSGS File

To change the following two prompts:

```
Delete col column header: Yes No
Delete row row header: Yes No
```

into their equivalents in French, would require the following VISIMSGS text file in RAM:

```
23 Qui Non
26 Supprime col
27 Supprime rang
```

If this VISIMSGS file existed in memory, then when either of the two prompts above was required by VisiCalc, they would appear as follows:

```
Supprime col column header: Qui Non
Supprime rang row header: Qui Non
```

Now if you press `[O]`, the deletion will occur, and if you press `[N]`, it will be cancelled.

Note that since the choice element in line 23 exists in RAM, it will be used with all other prompts that require a yes or no response.

# Setting the Video Display Parameters

## Contents

Introduction .....	186
Setting the Video Length .....	186
Setting the Video Width .....	186

## Introduction

The video display length and width can be controlled by keywords provided in the VisiCalc module. These keywords—`VLENGTH` and `VWIDTH`—affect the operation of a video display interface only when VisiCalc is running. When VisiCalc is not running, the HP-75 `WIDTH` command controls the display width, and the display length is determined by the video interface. Both of the VisiCalc video display keywords are executed from HP-75 `EDIT` mode.

## Setting the Video Length

To change the VisiCalc video display length from its default setting of 16, to any value between 1 and 255 lines, execute the VisiCalc `VLENGTH` command. `VLENGTH` returns error 89—`bad parameter`—if you specify a value that is not between 1 and 255.

`VLENGTH numeric expression`

**Examples:**

- `vlength 24`
- Sets the VisiCalc video display length to 24 lines.
- `vlength 16`
- Sets the video display length back to the default setting of 16 lines.

## Setting the Video Width

To change the VisiCalc video display width from its default setting of 32, to any value between 1 and 255 characters, execute the VisiCalc `VWIDTH` command. `VWIDTH` returns error 89—`bad parameter`—if you specify a value that is not between 1 and 255.



`VWIDTH` *numeric expression*

### Examples:

<code>vwidth 80</code>	Sets the VisiCalc video display width to 80 characters.
<code>vwidth 32</code>	Sets the display width back to the default setting of 32.

The HP-75 `WIDTH` command has no effect on the video display *while VisiCalc is running*, and the current `WIDTH` setting will be unchanged when you exit VisiCalc and while an extension function is running.

The values that you can set for the length and width will depend on which video interface you use. Also, although a particular interface may allow a certain parameter setting, the resulting display may be unusable. For example, you can specify a `VWIDTH` greater than 32 for the HP 82163 Video Interface, but if you display a worksheet using this setting, each row of the worksheet would require 2 or more display lines and the column headers might scroll off the top of the display. Refer to the owner's manual for your video interface or display device to determine the appropriate values for the length and width parameters.

Once set, the display length and width parameters will remain unchanged as long as the VisiCalc module is installed, or until you change them with another `VLENGTH` or `VWIDTH` command.

## Glossary

### Contents

Special Symbols .....	188
Terms .....	189

### Special Symbols

@	Placed at the beginning of an entry to indicate that the following entry is a value entry. See also <b>value</b> .
^	Caret character. <ul style="list-style-type: none"> <li>• Used to separate the column and row headers of user-defined coordinates.</li> <li>• Used as the exponential operator in formulas. See also <b>arithmetic operator</b>.</li> </ul>
[, ]	Used to delimit user-defined coordinates in a formula or function. See also <b>cell reference</b> , <b>delimiter</b> .
/-	Appears following the coordinates in the cell coordinates field during entry to indicate that a repeating label is being entered.
ERROR	Placed in a cell by VisiCalc to indicate that an arithmetic error has occurred. (May also be placed in a cell, as a function, by the user.) See also <b>function</b> , <b>VisiCalc function</b> .
<L>	Appears following the coordinates in the cell coordinates field during entry to indicate that a label is being entered. Also appears in the cell type field of the status line to indicate that the current cell contains a label. See also <b>status line</b> .
NA	"Not Available." Placed in a cell by VisiCalc when the value needed to calculate a particular cell is not available. See also <b>function</b> , <b>VisiCalc function</b> .
<V>	Appears following the coordinates in the cell coordinates field during entry to indicate that a value is being entered. Also appears in the cell type field of the status line to indicate that the current cell is a value or formula. See also <b>status line</b> .
vc	The keyword that is executed from HP-75 EDIT mode to enter the VisiCalc program.
W	Appears in an HP-75 file catalog entry to indicate that the file is a VisiCalc worksheet file.

## Terms

### A

---

**active boundaries:** The column and row bounding the area of the worksheet that contains nonempty cells.

**active worksheet:** The worksheet specified in the worksheet name template at the beginning of the current VisiCalc session. See also **worksheet**.

**active worksheet area:** The portion of the worksheet, bounded by the active boundaries, that contains cell entries.

**alternate viewing window:** An alternate access point, defined by the alternate window access cell, that provides a means to rapidly switch the display between two separate cells. Also called the alternate window. See also **primary window**.

**alternate window access cell:** The cell that will be displayed whenever the FET key is pressed from the primary window. The access point to the alternate viewing window, designated by the SHIFTFET keystroke.

**arithmetic hierarchy:** The prescribed order in which the HP-75 performs mathematical operations in an expression.

**arithmetic operator:** A symbol, or keyword, placed in a formula to cause an arithmetic operation to be performed. Arithmetic operators available to VisiCalc are: +, -, \*, /, DIV or \ (CTL/), and ^.

**ASCII:** American Standard Code for Information Interchange. The standard used by the HP-75 to represent its character set internally. Each HP-75 character corresponds to a decimal code of 0 through 255.

**automatic recalculation:** The process whereby the entire worksheet is recalculated after each cell is entered. See also **manual recalculation**, **recalculation**.

**Automatic Recalculation mode:** The worksheet condition under which automatic recalculation occurs. See also **Manual Recalculation mode**.

### B

---

**BASIC:** Beginner's All-purpose Symbolic Instruction Code, the programming language of the HP-75.

**boundaries:** In a worksheet, the columns and rows that bound a particular functional area of the worksheet. See also **active boundaries**, **physical boundaries**.

---

C

---

- calling cell:** When referring to functions, the cell that called the function. See also **function call**.
- card file:** A BASIC, text, keys, appointment, LEX, VisiCalc worksheet, or interchange (LIF) file that resides on a magnetic card or cards.
- cell:** The intersection of a column and a row in a worksheet.
- cell contents:** See **cell entry**.
- cell contents field:** The portion of the cell display that contains the current cell entry.
- cell coordinates:** A column header and a row header that specify a column-row position in the worksheet. Cell coordinates can be default, user-defined, or mixed coordinates.
- cell coordinates field:** The portion of the cell display that contains the cell coordinates.
- cell cursor:** The inverse video block that marks the location of the current cell in a video worksheet display.
- cell display:** The physical display of the current cell, as it appears in the 32-character display window of the HP-75.
- cell entry:** The value, formula, or label that is contained in a cell.
- cell reference:** Cell coordinates that refer to other cells.
- cell type field:** A portion of the status line. Shows whether the current cell is a value, label, or repeating label.
- circular reference:** A cell reference that can be traced, through one or more intermediate references, back to itself. Such a reference can never be recalculated, since its values depend indirectly on itself.
- column:** A vertically arranged collection of cells, headed by a column header.
- column header:** A name used to identify a specific column of the worksheet.
- column width:** A value between 0 and 255 that specifies the number of characters per column that will be displayed in the video display or printed by the Print command.
- command:** A keystroke or keystroke sequence, accessed by first pressing the slash key, that causes VisiCalc to perform certain operations.
- Command level:** The operational level of VisiCalc in which commands can be executed.

**control character:** A special display character—whose ASCII decimal code is 0 through 31 or 127—that may be interpreted by the HP-75 as an instruction to control the operation of the HP-75 display or an HP-IL peripheral device.

**control key:** The **CTL** key is used to access ASCII control characters and a number of special HP-75 display characters. It works like a shift key in that it is pressed together with some other key to access the corresponding control character.

**Coordinate Display mode:** The worksheet condition that determines whether cell coordinates are displayed as user-defined or default coordinates. See also **Default Coordinate Display mode**, **User-Defined Coordinate Display mode**.

**coordinates:** See **cell coordinates**.

**current cell:** The currently displayed worksheet cell in the primary window. Also, the cell to which the display will return when **FET** is pressed from the alternate window. The cell marked in the video worksheet display by the cell cursor.

**current worksheet:** See **active worksheet**.

**cursor:** A blinking symbol that appears in the display window to mark the current editing position on the entry line. The replace cursor (■) lets you type over characters; the insert cursor (⚡) inserts characters as you type. See also **cell cursor**.

## D

---

**data file:** A BASIC file that consists entirely of HP-75 BASIC **DATA** statements.

**Default Coordinate Display mode:** One of two coordinate display modes—the worksheet condition in which cell coordinates are displayed as default coordinates.

**default coordinates:** Cell coordinates that are specified by default headers.

**Default format:** A Local format in which the format of the cell entry defaults to the current global format.

**default headers:** Headers provided by VisiCalc automatically if there are no user-defined headers defined for the column or row. Column headers can be one or two letters, from A to IU, and row headers can be the integers 1 to 255.

**defaults:** Values provided by VisiCalc automatically (such as the column width) when no value has been specified by the user. The conditions, modes, or formats that exist in a new worksheet before any changes have been made by the user.

**delimiter:** A character (for example, ", [, and .) used to set off other characters or to separate parameters.

**device code:** A one- or two-character abbreviation that specifies one of the devices in an HP-IL system. May be one or two letters, a letter and digit, or a digit and letter.

**display:** The HP-75's display window.

**Dollars format:** A Local or Global format in which the values in a cell entry are rounded to two decimal places and are right-justified.

## E

---

**edit cursor:** See **cursor**.

**editing key:** A key or keystroke combination that controls the appearance and contents of an entry.

**Edit level:** The operational level of VisiCalc in which cell entries can be accessed by the **[EDIT]** key, or by pressing one of the keys that begin a value or label entry.

**ellipsis:** A series of three periods with no intervening spaces ( . . . ). Used to link the beginning and ending cell coordinates of a range.

**entry:** See **cell entry**.

**ERROR:** Placed in a cell by VisiCalc to indicate that an arithmetic error has occurred. (May also be placed in a cell, as a function, by the user.) See also **function**, **VisiCalc function**.

**escape character:** The control character (ASCII decimal code, 27) generated by the **[CTL][BACK]** keystroke combination and by the **CHR\$(27)** function.

**escape code:** A sequence of characters, the first being the escape character, that may be interpreted by the HP-75 as an instruction to control the operation of the HP-75 display or an HP-IL peripheral.

**escape keystroke:** **[CTL][BACK]**

**escape sequence:** See **escape code**.

**exit message:** The message displayed by VisiCalc when the program is exited. It consists of the message **Leaving worksheet:**, followed by the file name of the current worksheet.

**expression:** A series of one or more operands separated by zero or more arithmetic operators.

**extension function:** A function that can be called from VisiCalc, and which is created by the user as a BASIC program. Parameters are passed to and from extension functions via the **VISIDATA** data file.

**external cell reference:** A cell reference to one or more cells in another worksheet.

---

**F**

---

**file name:** A unique string of 1 to 8 letters or numbers, beginning with a letter or a period, that identify an HP-75 file.

**first cells:** The cells, at the beginning of each column or row of the target range, that will receive a copy of the first cell in the source range when a Replicate command is executed.

**foreign language prompts:** The various messages and prompts displayed by VisiCalc, translated into another language. See also `VISIMSGS`.

**format:** A specification that determines the way in which the contents of a cell are displayed.

**Formatted Display mode:** The worksheet condition in which the cell display will show values and results in Integer or Dollars format if one of those two formats have been set for that cell (either locally or globally).

**formula:** A cell entry that can contain values, cell coordinates, and functions connected by operators.

**Formula Display mode:** The worksheet condition in which the formulas, in any cell with a formula entry, are displayed in the cell display.

**forward reference:** A reference to a cell in a column or row farther to the right or below the cell containing the reference.

**function:** A built-in routine that operates on zero or more parameters to produce a single numeric value. See also **HP-75 function**, **VisiCalc function**, **extension function**.

**function call:** A cell entry, or part of a cell entry, that consists of a function name followed by a parameter list.

---

**G**

---

**General format:** A Local or Global format in which values are right-justified, labels are left-justified, and as many digits of the value are displayed as possible.

**global column width:** The single width specified for all columns in the worksheet. Local column widths override the global column width.

**Global format:** A format that controls the way the contents of the cells are displayed on the entire worksheet.

**Graphing format:** A Local or Global format in which cell values are converted to asterisks.



---

**H**


---

**header:** The symbol or character string that identifies a particular column or row of a worksheet. See also **default headers**, **user-defined headers**.

**hierarchy:** See **arithmetic hierarchy**.

**HP-75 function:** A function contained in the operating system of the HP-75. Most HP-75 numeric functions are accessible from VisiCalc. All string functions and functions that return a string are *not* accessible from VisiCalc.

**HP-IL:** The Hewlett-Packard Interface Loop. A system of peripheral devices controlled by the HP-75 or any HP-IL controller.

---

**I**


---

**inconsistent worksheet:** A worksheet where the calculated results do not reflect the current data in the worksheet. See also **circular reference**.

**integer:** A number that is not a fraction and does not include a fraction.

**Integer format:** A Local or Global format in which values and results are rounded to an integer.

**interface:** The hardware that connects a controller and peripherals and enables them to function together. Also the relationship between two processes or entities (such as, the interface between VisiCalc and the video display).


---

**L**


---

**label:** A cell entry beginning with a letter, single or double quotes, ^, #, \$, %, &, or a space. Label entries are used to identify or describe other cell entries.

**label entry:** A cell entry that consists of a label. See also **cell entry**.

**Language Extension (LEX) file:** A special program file available on mass storage media and in plug-in ROMs that may define new keywords and extend computer capabilities. The  keyword is in a LEX file.

**left-justified:** A condition in which the left-hand character of an item occupies the left-hand position of the space allotted for the item.

**Left-Justified format:** A Local or Global format in which values or labels are displayed left-justified in the column.



**left physical boundary:** The left-most column of the active worksheet. Also column A of the worksheet. See also **boundaries**.

**list:** A series of items, called list elements, separated by commas and enclosed in parentheses. List elements may be any of: a numeric constant, a cell coordinate, an external coordinate, a range, an external range, a function (including VisiCalc, HP-75, LEX, or extension functions), or an expression. A list may contain up to 255 list elements.

**local column width:** The width specified for a single column. Every column in a worksheet may have a separate local column width.

**Local format:** A format that controls the way the contents of a single cell are displayed. Each cell in the worksheet may have a local format specified for it.


**local format field:** Part of the status line. Contains a letter specifying the local format of the current cell.

**logical operators:** A keyword placed in a formula to cause a logical operation to be performed. Logical operators are: AND, OR, EXOR, NOT.

## M

---

**magnetic card:** A medium on which information is stored. See also **card file**.

**manual recalculation:** The process whereby the entire worksheet is recalculated only when the  key is pressed at Top level. See also **automatic recalculation**.

**Manual Recalculation mode:** The worksheet condition in which the worksheet is recalculated by manual recalculation. See also **Automatic Recalculation mode**.

**mass storage device:** An I/O device like the card reader or HP 82161A Digital Cassette Drive that may be used to copy files from memory to mass storage media and from mass storage media to memory.

**mathematical hierarchy:** See **hierarchy**.

**memory:** The number of bytes available in the HP-75 for worksheet or other file storage.

**mixed coordinates:** Cell coordinates that contain both default and user-defined headers. See also **default coordinates**, **user-defined coordinates**.

N

---

**NA:** “Not Available.” Placed in a cell by VisiCalc when the value needed to calculate a particular cell is not available. See also **function**, **VisiCalc function**.

**name field:** The part of the worksheet name template that contains or can contain the worksheet file name.

**numeric expression:** See **expression**.

O

---

**operand:** A number, a cell reference, or a function call that appears in an expression.

**operational levels:** The three levels of VisiCalc: Top level, Edit level, and Command level.

**operator:** A symbol or keyword that combines or compares the values of two operands or expressions. Arithmetic, relational, and logical operators result in a numeric value.

**overflow display:** A display, consisting of a series of “greater than” symbols (>), which appears in the video display or printed worksheet to indicate that the value in the cell cannot be displayed in the available column width.

P


---

**parameter:** A list element that appears in a parameter list as one of the inputs of a function call. A value that determines the status of a worksheet condition or state (such as the value of the global column width). See also **defaults**, **list**.


**parameter list:** A list used to pass parameters to a function. See also **list**.

**peripheral:** An external HP-IL device that is controlled by the HP-75.

**physical boundaries:** The columns and rows that are at the physical limits of worksheet size. The physical row boundaries are row 1 and 255. The physical column boundaries are column A and IU. See also **active boundaries**.

**precalculation:** The process whereby an expression is evaluated (by pressing ) before it is entered into a cell and only the result of that calculation is placed in the cell.

**precision:** A number of significant digits with which the HP-75 stores a numeric value.

**primary window:** The normal worksheet display window. Accessed from the alternate viewing window by the  keystroke. See also **alternate viewing window**.

**printer boundary:** The area of the worksheet, specified by the upper-left and lower-right cells, to be printed.

- printer setup string:** A character string, specified during execution of the Print command, that provides a way to transmit control codes, escape sequences, and printable title strings to the printer.
- printer width:** The number of characters per line that the printer will print before generating a carriage-return/line-feed.
- print options menu:** The first prompt of the Print command, which allows you to specify the type of printout to be produced.
- prompt:** A VisiCalc message that specifies the available options for the next operation, or that describes the next required entry.

## R

---

- random access memory (RAM):** The circuits in the HP-75 that store user-created programs, data, text, appointments, worksheets, variable values, and related information. RAM may be changed or erased by the user at any time.
- range:** A contiguous series of cells, all in the same column or row. Specified by the coordinates of the first and last cells in the range, separated by an ellipsis (. . .). See also **ellipsis**, **cell coordinates**.
- read-only memory (ROM):** Permanent memory that cannot be changed or erased.
- recalculation:** The process of calculating new values for a cell or all the cells in the worksheet.
- recalculation order:** The order in which formulas in cells of a worksheet are recalculated—either column-by-column or row-by-row.
- relational operators:** A symbol placed in a formula to cause a relational operation to be performed. Relational operators are: =, <, <=, >, >=, and <> (or #).
- relative replication:** A type of replication in which the cell references in formulas in the target range are made relative to their new locations in the worksheet.
- remaining memory field:** A part of the status line that shows remaining user memory.
- repeating label:** A label consisting of a character or group of characters that VisiCalc will automatically repeat to fill the cell.
- replication:** The process of repeating an entry across a column or row, or repeating an entire column or row on the worksheet.
- Result Display mode:** A worksheet condition in which VisiCalc displays the results of any cell entry in the worksheet.

**right-justified:** A condition in which the right-hand character of an item occupies the right-hand position of the space allotted for the item.

**Right-Justified format:** A Local or Global format in which values or labels are displayed right-justified in the column.

**row:** A horizontally arranged collection of cells, headed by a row header.

**row header:** The name used to identify a specific row of the worksheet.

**row header width:** The column width of the row header column.

## S

---

**scientific notation:** A short-hand system to express numbers too large or too small to fit the display—that is, numbers that can't be expressed adequately with 12 digits.

**simple numeric variable:** A name (a single letter or a letter-digit combination) representing a location in memory that stores a numeric value. A simple numeric variable can be used interchangeably with the number it references.

**source range:** The cell or cells to be replicated.

**spreadsheet:** See **worksheet**.

**start-up:** The condition of VisiCalc immediately after the `W` keyword is executed. The process of entering VisiCalc and specifying a file name.

**status line:** A single line display of information about the worksheet—accessed by the `[SHIFT][EDIT]` keystroke.

## T

---

**target range:** The cell or cells into which source range cells are to be replicated.

**toggle:** One of several keys or keystroke combinations that change the state of one of VisiCalc's display modes, or that alternate the cell display with some other display. Toggles include: `[TIME]`, `[SHIFT][TIME]`, `[APPT]`, `[SHIFT][APPT]`, `[SHIFT][EDIT]` and `[FET]`.

**Top level:** The level to which VisiCalc returns after any operation. The principal activity of this VisiCalc level is to view the worksheet. See also **Command level**, **Edit level**.

**top physical boundary:** The top row of the active worksheet. Also row 1 of the worksheet. See also **boundaries**.

## U

---

**Unformatted Display mode:** The worksheet condition in which numbers are displayed to the full precision of the HP-75. See also **Formatted Display mode**.

**User-Defined Coordinate Display mode:** One of two Coordinate Display modes. The worksheet condition in which cell coordinates are displayed as user-defined coordinates. See also **Default Coordinate Display mode**.

**user-defined coordinates:** Cell coordinates that are specified by user-defined headers.

**user-defined headers:** Alphanumeric labels that the user can create to replace the original default column and row headers.

## V

---

**value:** A number in a cell.

**value entry:** A cell entry that consists of a value or formula. See also **cell entry**.

**video display:** The display that appears on a video display device.

**video display device:** An HP-IL output device whose device code has appeared in a `DISPLAY IS` declaration.

**video interface:** A device that enables you to use a video display device with VisiCalc. See also **interface**.

**VisiCalc function:** A function built-in to VisiCalc. See also **function**.

**VISIDATA:** An HP-75 data file used to pass parameters to an extension function.

**VISIMSGS:** An HP-75 text file that can be created to contain the VisiCalc prompts and messages. If `VISIMSGS` exists in memory, VisiCalc uses the prompts and messages in `VISIMSGS` in place of those built into VisiCalc. See also **foreign language prompts**.

## W

---

**window:** The currently viewable part of a worksheet. The HP-75 display is a single-cell window. A video display device provides a multi-cell window. See also **alternate viewing window**, **primary window**.

**working message:** The `Working` message, which appears in the display whenever VisiCalc is engaged in an operation that will cause a perceptible delay in response.

- worksheet:** A grid consisting of columns lettered A, B, C, and so on, and rows numbered 1, 2, 3, and so on. At each intersection of a column and a row, there is a cell where information such as a number, a formula, or a message can be entered.
- worksheet catalog:** In the worksheet name template, the sequential listing of the names of worksheet files in memory.
- worksheet file:** An HP-75 file that contains a worksheet. Defined as a type W file. Can be CATALOGued, COPYed, PURGED, RENAMEd but not LISTed, PLISTed, RUN, CALLED, EDITed, RENUMBERed, or TRANSFORMed.
- worksheet file name:** A standard HP-75 file name given to a worksheet file. Also called worksheet name.
- worksheet name template:** The display Enter worksheet name: ■ that appears at start-up, and which may contain a worksheet file name.
- worksheet status line:** See status line.

# Command and Operation Summary and Index





## Contents

Start-Up Keys .....	201
Top Level Keys .....	202
Edit Level Keys .....	203
Command Level .....	204
Delete Command .....	204
Format Command .....	205
Global Command .....	205
Header Command .....	206
Insert Command .....	208
Move Command .....	208
Print Command .....	208
Replicate Command .....	209
Video Command .....	210
Width Command .....	210
Repeating Label Command .....	210

Page numbers following an entry indicate the page in the manual where the principal discussion of the key operation or command is located.

## Start-Up Keys

The following key operations are active when the start-up message is displayed:

-  Calls the file name of the next worksheet in memory to the worksheet name template. **(Page 25.)**
-  Calls the file name of the previous worksheet in memory to the worksheet name template. **(Page 25.)**
-  Calls the file name of the least recently accessed worksheet in memory to the worksheet name template. **(Page 25.)**
-  Calls the file name of the most recently accessed worksheet in memory to the worksheet name template. **(Page 25.)**



- CLR** Clears the currently displayed worksheet file name from the worksheet name template. **(Page 26.)**
- RTN** Selects the displayed worksheet as the active worksheet and switches VisiCalc to Top level. If the worksheet does not exist, it is created. An invalid or non-worksheet file name causes the HP-75 to beep and VisiCalc to move the cursor to the first character in the filename. **(Page 25.)**
- ATTN** Exits VisiCalc and returns to HP-75 EDIT mode. **(Page 25.)**
- SHIFT** **ATTN** Exits VisiCalc and returns to HP-75 EDIT mode. **(Page 25.)**

All HP-75 display editing keys are active to edit the file name entry.

## Top Level Keys

The following key operations are active at Top level:

- ATTN** Returns to Top level. **(Page 37.)**
- SHIFT** **ATTN** Exits VisiCalc. **(Page 47.)**
- TIME** Toggles the cell coordinates field of the cell display between User-Defined and Default Coordinate Display modes. **(Page 62.)**
- SHIFT** **TIME** Toggles the cell contents field of the cell display between User-Defined and Default Coordinate Display modes. **(Page 62.)**
- APPT** Toggles the worksheet between Formula and Result Display modes. **(Page 62.)**
- SHIFT** **APPT** Toggles the cell display between Formatted and Unformatted Display modes. **(Page 63.)**
- SHIFT** **EDIT** Toggles the display between the current cell and the worksheet status line. **(Page 30.)**
- FET** Toggles the display between the current cell and the window access cell of the alternate viewing window. **(Page 34.)**
- SHIFT** **FET** Defines the alternate window access cell. **(Page 34.)**
- ↑, ↓, ←, →** Moves the cell display one cell up, down, left, and right, respectively. **(Page 32.)**
- SHIFT** **↑, ↓, ←, →** Moves the cell display to the top and left physical boundaries, and to the bottom and right active boundaries of the worksheet, respectively. **(Page 32.)**



<b>[RUN]</b>	Moves the cell contents of the cell display one character position to the left. <b>(Page 29.)</b>
<b>[SHIFT] [RUN]</b>	Moves the cell contents of the cell display one character position to the right. <b>(Page 29.)</b>
<b>[TAB]</b>	Moves the cell contents of the cell display one window width to the left. <b>(Page 29.)</b>
<b>[SHIFT] [TAB]</b>	Moves the cell contents of the cell display one window width to the right. <b>(Page 29.)</b>
<b>[!]</b>	Recalculates the entire worksheet. <b>(Page 80.)</b>
<b>[@], [+], [-], [*], [()], [()], or digit</b>	Switches VisiCalc to Edit level, ready for a value entry. <b>(Page 41.)</b>
<b>['], ["], [^], [#], [\$], [%], [&amp;], letter, or space</b>	Switches VisiCalc to Edit level, ready for a label entry. <b>(Page 39.)</b>
<b>[EDIT]</b>	Switches VisiCalc to Edit level, ready to change the current cell contents. <b>(Page 58.)</b>
<b>[CLR]</b>	Clears the contents of the current cell and switches VisiCalc to Edit level, ready for a new entry of the same type as the just-cleared entry. <b>(Page 38.)</b>
<b>[SHIFT] [&gt;]</b>	Initiates the Go To command. <b>(Page 33.)</b>
<b>[/]</b>	Switches VisiCalc to Command level. <b>(Page 46.)</b>

## Edit Level Keys

The following key operations are active at Edit level:

<b>[ATTN]</b>	Cancels the current cell entry, restores the previous entry, and returns to Top level. <b>(Page 38.)</b>
<b>[SHIFT] [ATTN]</b>	Exits VisiCalc. <b>(Page 47.)</b>
<b>[TIME]</b>	Toggles the cell coordinates field of the cell display between User-Defined and Default Coordinate Display modes. This toggle setting is stored with the worksheet. <b>(Page 62.)</b>
<b>[→]</b>	Moves the edit cursor one character position to the right. <b>(Page 58.)</b>
<b>[←]</b>	Moves the edit cursor one character position to the left. <b>(Page 58.)</b>
<b>[SHIFT] [→]</b>	Moves the edit cursor one position past the last character position of the current entry. <b>(Page 58.)</b>
<b>[SHIFT] [←]</b>	Moves the edit cursor to the first character position of the current entry. <b>(Page 58.)</b>
<b>[CTL] [→]</b>	Right-shifts the edit cursor one window width or to the end of the cell contents field, whichever distance is shorter. <b>(Page 58.)</b>

<b>CTL</b> <b>←</b>	Left-shifts the edit cursor one window width or to the beginning of the cell contents field, whichever distance is shorter. <b>(Page 58.)</b>
<b>I/R</b>	Exchanges the replace cursor (■) with the insert cursor (✱). Characters that are entered after this keystroke will be inserted to the left of the cursor. <b>(Page 58.)</b>
<b>SHIFT</b> <b>I/R</b>	Causes the next keystroke or keystroke combination to display the character associated with it, regardless of key redefinitions. <b>(Page 58.)</b>
<b>DEL</b>	Deletes the character under the cursor and moves the remaining characters to the left. <b>(Page 58.)</b>
<b>SHIFT</b> <b>DEL</b>	Deletes the character under the cursor and all characters to the right of the cursor. <b>(Page 58.)</b>
<b>CLR</b>	Clears the cell contents field. <b>(Page 58.)</b>
<b>!</b>	Evaluates a formula, during value entry, before it is stored in a cell. The formula will be over-written by the result and cannot be recovered if <b>RTN</b> is pressed. <b>(Page 79.)</b>
<b>BACK</b>	Replaces the character to the left of the edit cursor with a blank, then moves the cursor one position to the left. In insert mode, the characters at and to the right of the cursor are moved one position to the left. <b>(Page 58.)</b>
<b>RTN</b>	Stores the cell entry and returns to Top level. <b>(Page 38.)</b>

## Command Level

- /** **D** Enters Command level from Top level. **(Page 46.)**

**Prompt:** Command: `DEFGHIMPRVW-`

## Delete Command

- /** **D** Initiates the Delete command. Allows deletion of a column or row. **(Page 66.)**

**Prompt:** Delete: `Col Row`

- D** **C** Selects deletion of the column in which the cell display lies.

**Prompt:** Delete col *current column* : `Yes No`

- Y** Deletes the selected column.  
**N** Cancels the delete column operation.

- D** **R** Selects deletion of the row in which the cell display lies.

**Prompt:** Delete row *current row* : `Yes No`

- Y** Deletes the selected row.  
**N** Cancels the delete row operation.

## Format Command

- ☒ ☐ **F** Initiates the Local Format command. Allows the display format of a particular cell to be set. (Page 132.)

**Prompt:** `Format (local): D G I L R $ *`

- ☐ ☐ **D** Sets the Local format to use the current global format.
- ☐ ☐ **G** Sets the Local format to the General format. Values are right-justified with as many digits displayed as possible. Labels are left-justified.
- ☐ ☐ **I** Sets the Local format to Integer format. Values are rounded to an integer and are right-justified without a decimal point. This command also alters the format of the cell display in the HP-75's display window when Formatted Display mode is set.
- ☐ ☐ **L** Sets the Local format to Left-Justified format. Causes value or label entries to be left-justified within the cell display.
- ☐ ☐ **R** Sets the Local format to Right-Justified format. Causes value or label entries to be right-justified within the cell display.
- ☐ ☐ **\$** Sets the Local format to Dollars format. Values are rounded to two decimal places and are right-justified. This command also alters the format of the cell display in the HP-75's display window when Formatted Display mode is set.
- ☐ ☐ **\*** Sets the Local format to Graphing format. Converts cell values to asterisks. The number of asterisks is equal to the absolute value in the cell rounded to an integer.

## Global Command

- ☒ ☐ **G** Initiates the Global command. (Page 138.)

**Prompt:** `Global: Col Order Recalc Format`

- ☐ ☐ **C** Allows you to set the column width of the entire worksheet to any width from 0 to 255 characters. (Page 139.)

**Prompt:** `Col width (global): current global column width`

Enter the selected Global Column Width and press .

- ☐ ☐ **O** Allows you to specify the recalculation order of the worksheet. (Page 80.)

**Prompt:** `Order of recalc: Col Row`

- ☐ **C** Sets the recalculation order to be column by column, beginning with column A.
- ☐ **R** Sets the recalculation order to be row-by-row, beginning with row 1.

- [G] [R]** Allows you to specify the Recalculation mode of the worksheet. **(Page 80.)**

**Prompt:** Recalc: Auto Manual

**[A]** Sets the worksheet to Automatic Recalculation mode.

**[M]** Sets the worksheet to Manual Recalculation mode.

- [G] [F]** Allows you to set the format of all cells in the worksheet except those that have had a local format set (with the Format command). The available global formats are the same as those available to the Local Format command, except that there is no Default format. **(Page 139.)**

**Prompt:** Format (global) G I L R \$ \*

**[G]** Sets the worksheet format to the General format. Values are right-justified with as many digits displayed as possible. Labels are left-justified.

**[I]** Sets the worksheet format to Integer format. Values are rounded to an integer and are right-justified without a decimal point. This command also alters the format of the cell display in the HP-75's display window when Formatted Display mode is set.

**[L]** Sets the worksheet format to Left-Justified format. Causes value or label entries to be left-justified within the cell display.

**[R]** Sets the worksheet format to Right-Justified format. Causes value or label entries to be right-justified within the cell display.

**[\$]** Sets the worksheet format to Dollars format. Values are rounded to two decimal places and are right-justified. This command also alters the format of the cell display in the HP-75's display window when Formatted Display mode is set.

**[\*]** Sets the worksheet format to Graphing format. Converts cell values to asterisks. The number of asterisks is equal to the absolute value in the cell rounded to an integer.

## Header Command

- [/] [H]** Initiates the Header command. **(Page 54.)**

**Prompt:** Headers: Col Row Width Suppress

- [H] [C]** Allows you to create or edit user-defined column headers. **(Page 54.)**

**Prompt:** Col header *current column* : (No existing user-defined header.)

Col header *current column* : *existing user-defined header*

The *current column* is the default header of the column where the cell display was located when the command was executed.

The *existing user-defined header* is any existing user-defined header for the current column.

You can perform the following operations at this point:

- To enter a new user-defined header, type the new header, then press **[RTN]** to store the header.
- To modify an existing header, press **[EDIT]**, use the editing keys to make the changes, then press **[RTN]** to store the header.
- To display other column headers, press **[←]**, **[→]**, **[SHIFT][←]**, or **[SHIFT][→]**.
- To return to Top level after all desired headers have been entered, press **[ATTN]**.

**[H] [R]** Allows you to create or edit user-defined row headers. **(Page 54.)**

**Prompt:** Row header *current row* : (No existing user-defined header.)

Row header *current row* : *existing user-defined header*

The *current row* is the default header of the row where the cell display was located when the command was executed.

The *existing user-defined header* is any existing user-defined header for the current row.

You can perform the following operations at this point:

- To enter a new user-defined header, type the new header, then press **[RTN]** to store the header.
- To modify an existing header, press **[EDIT]**, use the editing keys to make the changes, then press **[RTN]** to store the header.
- To display other row headers, press **[↑]**, **[↓]**, **[SHIFT][↑]**, or **[SHIFT][↓]**.
- To return to Top level after all desired headers have been entered, press **[ATTN]**.

**[H] [W]** Allows you to set the width of the row header column to any value between 0 and 255 characters. **(Page 55.)**

**Prompt:** Width row headers: (No current row header width.)

Width row headers: *current row header width*

Enter the selected row header width and press **[RTN]**, or press **[CLR] [RTN]** to reset the header width to the global column width.

**[H] [S]** Allows you to suppress default headers in printed listings and on a video display. **(Page 55, 107.)**



**Prompt:** Suppress default headers: Yes No

**[Y]** Suppresses default headers.





**[N]** Restores default headers to video display or printed listing.



## Insert Command

-   Initiates the Insert command. (Page 64.)



**Prompt:** Insert: Col Row

-   Inserts a column to the left of the column in which the cell display lies.
-   Inserts a row just above the row in which the cell display lies.


## Move Command



-   Initiates the Move command. (Page 67.)

**Prompt:** Move: Col Row


-   Moves the column in which the cell display lies to the position currently occupied by the specified destination column.

**Prompt:** Move col *current column* to: ■

Enter the destination column and press .



-   Moves the row in which the cell display lies to the position currently occupied by the specified destination row.

**Prompt:** Move row *current row* to: ■







Enter the destination row and press .

When moving a column (or row), all columns (or rows) between the source and the destination, including the destination, will move one position toward the source, and the source will be placed in the destination's previous location.

## Print Command

-   Initiates the Print command and allows you to select the print option. (Page 106.)

**Prompt:** Pr: Worksheet +Headers Formulas

-   Allows you to print the worksheet without headers.
-   Allows you to print the worksheet-plus-headers.
-   Allows you to print the worksheet formulas.

After you select the type of printout, VisiCalc prompts for the boundaries of the worksheet area to be printed. **(Page 109.)**

**Prompt:** Upper left: *cell coordinates*

Press **[RTN]** to use the default boundary or (for a different boundary) press **[CLR]**, enter the new coordinates, then press **[RTN]**. **(Page 109.)**

**Prompt:** Lower right: *cell coordinates*

Press **[RTN]** to use the default boundary or (for a different boundary) press **[CLR]**, enter the new coordinates, then press **[RTN]**. After you specify the print boundaries, VisiCalc prompts for an optional printer setup string. **(Page 110.)**

**Prompt:** Printer setup: ■ (No current printer setup string.)

Printer setup: *current printer setup string*

Enter any combination of escape sequences, control codes, and printable characters (up to 95 characters), then press **[RTN]**. After you specify a setup string, VisiCalc prompts for the printer width. **(Page 112.)**

**Prompt:** Printer width: *current HP-75 PWIDTH setting*

Press **[RTN]** to use the current PWIDTH setting or (to enter a new printer width) press **[CLR]**, enter the new width, then press **[RTN]**. VisiCalc will respond with the *Working* message and will begin to print the worksheet (or formulas). The *Working* message will be displayed until the printout is complete.

If you attempt to execute the Print command without a correctly connected and assigned printer on the loop, the HP-75 will beep and VisiCalc will remain at the command prompt line.

## Replicate Command

**[/][R]** Initiates the Replicate command. **(Page 85.)**

**Prompt:** Repl source: *current cell* ■

- To use the displayed cell as the entire source range, press **[RTN]**.
- To specify a range of cells (beginning with the current cell) as the source range, enter an ellipsis, then the coordinates of another cell, followed by **[RTN]**.
- To specify a range of cells that begin with a different cell than the current cell, press **[CLR]**, enter a source range beginning with a cell different than the one displayed, then press **[RTN]**.

After you enter the source range, VisiCalc prompts for the target range.

**Prompt:** Repl target: ■

Enter the target range and press **[RTN]**. Before pressing **[RTN]**, you can review the source range by pressing **[↑]**; press **[↓]** to return to the target range prompt. If the cell or cells being replicated contain cell references, VisiCalc will prompt for the type of replication.

**Prompt:** Repl: No Relative Both

- ☐ **N** Replicates the source range into the target range with exactly the same cell references as those in the source range cells.
- ☐ **R** Replicates the source range into the target range with all relative cell references.
- ☐ **B** Allows you to specify No change or Relative for every cell reference in the source range.

**Prompt:** *source range coordinates* : underlined cell reference +No Rel

For each cell in the source range (*source range coordinates*) that contains a cell reference, each cell reference in that cell's entry will be displayed in turn (*cell reference*) with the +No Rel prompt.

- ☐ **N** Causes that particular cell reference to be replicated with no changes.
- ☐ **R** Causes that particular cell reference to be replicated as a relative reference.

## Video Command

- ☒ ☐ **V** Initiates the video command. **(Page 120.)**

**Prompt:** Video active: Yes No

- ☐ **Y** Clears the video display and activates it if it is not active.
- ☐ **N** Deactivates the video display if it is active.

If you attempt to execute the Video command without a correctly connected and assigned video display device on the loop, the HP-75 will beep and VisiCalc will remain at the command prompt line.

## Width Command

- ☒ ☐ **W** Initiates the Local Column Width command, which allows you to set the width of the current column to any value between 0 and 255 characters. **(Page 136.)**

**Prompt:** Width of col *current column* : █ (No current column width.)

Width of col *current column* : *current column width*

Enter the selected column width and press , or press   to reset the column width of the *current column* to the global column width.

## Repeating Label Command

- ☒ ☐ **-** Initiates the Repeating Label command. **(Page 61.)**

**Prompt:** *cell coordinates* (/-): █

Any entry is treated as a label and when the cell is viewed on a video display device or printed, the entire cell is filled with repetitions of the character or characters entered in the cell.



# Subject Index

Page numbers in **bold** type indicate primary references; page numbers in standard type indicate secondary references.

## A

---

@ sign, **44**, **72**, 78  
Accessing a cell, **32**  
Accessing other worksheets, 46, **143**  
Active boundary, **27**, 32  
Active worksheet, **27**  
Active worksheet area, **27**, 109  
Alternate viewing window, **34**, 35  
Alternate window access cell, **34**  
**APPT** (*Formula/Result toggle*) key, **21**, 22, 23, 42, **62**, 122  
Arithmetic  
    hierarchy, **71**  
    operators, 41, **70**  
Arrow keys, 21, 25, 27, **32**, 37, **58**, 86, 126  
ASCII, **39**, 110  
Asterisk, **133**, **140**  
At sign (@), **44**, **72**, 78  
**ATTN** (*attention*) key, 25, **27**, 31, **37**, 41, **47**, 57, **58**, 86, 112, 165  
Automatic recalculation, **80**  
Automatic Recalculation mode, **80**  
AVERAGE (*function*), 45, **76**, 175

## B

---

**BACK** key, **58**  
Blanking a cell, **10**, **58**  
Both option in replication, **93**  
Boundary,  
    active, **27**, 32  
    left physical, **27**, 32  
    lower-right, **109**  
    physical, **27**, 32  
    printer, **109**  
    top physical, **27**, 32  
    upper-left, **109**  
Brackets, use of, **30**, **38**, 40, **54**, 61

## C

---

Calling cell, **152**, 153, 156, 158  
Caret (^), use of, **30**, **38**, **70**, **71**  
Card file, **103**  
Cell,  
    accessing a, **32**  
    blanking a, **10**, **58**  
    clearing a, **10**, **58**  
    window access, **34**  
Cell contents, **28**  
Cell contents, editing, **59**  
Cell contents field, **21**, 23, 28, **29**, 62  
Cell coordinates, **21**, 27  
Cell coordinates field, **21**, 23, **28**, 62  
Cell cursor, **126**, 129  
Cell display, **20**, **28**  
    modifying the, **61**  
    scrolling the, 23, **29**  
Cell entry,  
    creating, 38, 39-41, 42, 44, **57**, 58  
    editing, **57**, 58, **59**  
Cell in another worksheet, **148**  
Cell reference, external, 46, **143**  
Cell type field, **31**  
Circular reference, **83**  
Clearing a cell, **10**, **58**  
**CLR** (*clear*) key, **25**, **38**, 57, **58**, 86, 109, 111, 136  
Colon, **21**, **29**  
Column,  
    adjusting width of, **136**, **139**  
    deleting a, **66**  
    inserting a, **64**  
    moving a, **67**  
Column header width, **136**, **138**  
Column width,  
    Global, **138**, **139**  
    Local, **136**, **138**

Command and Operation Summary, **201-210**

Command level **36**, **46**

Commands

Delete, **46**, **66**

Format, **46**, **132**, **135**

Global, **46**, **80**, **138**

Header, **30**, **46**, **54**, **123**, **125-126**, **128**

Insert, **46**, **64**

Move, **46**, **67**

Print, **46**, **106**

Repeating Label, **47**, **61**

Replicate, **46**, **85**, **135**, **140**

Video, **47**, **120**

Width, **47**, **122**, **128**, **136**, **138**, **139**

Computer memory, **31**, **47**

Control code, **39**, **106**, **110**

Control key, **110**

Coordinate Display mode, **62**

Coordinates,

default, **28**, **42**

mixed, **28**, **42**

user-defined, **28**, **30**, **42**

Creating

header, **54**

worksheet, **20**, **24**, **25**

Current cell, **29**, **30**, **86**, **123**

Current cell field, **31**

Cursor, **24**, **39**

## D

*/D. See Delete command*

Data file, **155**, **156**, **157**, **160**

Dealer information, **174**

Default

Coordinate Display mode, **62**

coordinates, **28**, **42**

format, **133**,

headers, **23**, **30**, **54**

**[DEL]** (*delete character*) key, **58**

Delete column or row, **66**

Delete command, **66**

Destination column or row, **67**

Device code, **105**, **106**, **120**

Display,

cell, **28**

modifying the cell, **61**

overflow, **140**

Display format, **63**

Display mode,

Default Coordinate, **62**

Formatted, **21**, **63**

Formula, **21**, **29**, **42**, **43**, **62**, **122**

Result, **62**

Unformatted, **63**

User-Defined Coordinate, **62**

Dollars format, **63**, **132**, **133**

## E

Edit cursor, **24**, **39**, **58**, **61**

Editing

cell contents, **59**

during cell entry, **57**

headers, **57**

with **[EDIT]**, **38**, **59**

with redefined keys, **61**

**[EDIT]** (*Edit level*) key, **38**, **39**, **57**, **59**, **67**

Edit level, **36**, **38**, **58**

Ellipsis, **73**, **86**

ERROR (*function*), **45**, **67**, **76**, **77**, **175**

Escape character, **110**, **111**

Escape keystroke, **110**

Escape sequence, **110**, **111**

Exclamation point key (**[!]**), **38**, **58**, **79**, **80**, **83**, **84**

Exit message, **24**, **39**, **47**

Exponentiation, **70**

Expression, **70**

Extension function, **45**, **152**

accessing an, **153**

creating an, **159**, **164**

passing values to and from an, **155**

External cell reference, **46**, **143**, **150**

## F

*/F. See Format command*

**[FET]** (*toggle alternate window*) key, **34**, **35**, **37**

Field,

cell contents, **21**

cell coordinates, **21**, **28**

cell type, **31**

current cell, **31**

global format, **31**

local format, **31**

recalculation mode, **31**

recalculation order, **31**

remaining memory, **31**

worksheet name, **31**

File,  
   Language Extension, **166**  
   worksheet, **24, 25, 26**  
 File name, **24, 25, 26**  
 First cells, **86, 87**  
 Foreign language prompts, **181**  
 Format command, **132, 135**  
 Format,  
   display, **63**  
   Dollars, **63, 132, 133**  
   General, **133**  
   Global, **122, 138**  
   Graphing, **133, 140**  
   Integer, **63, 133**  
   Left-Justified, **133**  
   Local, **122, 132**  
   Local Graphing, **133, 140**  
   Replicating a local, **135**  
   Right-Justified, **133**  
 Formatted Display mode, **63**  
 Formula, **29, 42**  
 Formula Display mode, **21, 29, 42, 43, 62, 122**  
 Formulas, rounding, **72**  
 Forward reference, **83**  
 Function,  
   extension, **152, 153, 155, 159, 164**  
   HP-75, **78, 175-177**  
   trigonometric, **175, 177**  
   VisiCalc, **45, 72, 175-177**  
 Function call, **45**

## G

---

/G. See *Global command*  
 General format, **133**  
 Global column width, **138, 139**  
 Global command, **80, 138**  
 Global format, **138, 139**  
 Global format field, **31**  
 Global order, **80, 138**  
 Global recalculation, **80, 138**  
 Glossary, **188-200**  
 Go To command, **22, 33, 38**  
 Graphing format, **133, 140**

## H

---

/H. See *Header command*  
 Header,  
   clearing a, **57**  
   column, **54**  
   creating a, **54**  
   default, **23, 30, 54**  
   editing a, **57**  
   row, **54**  
   suppressing the, **55, 108, 126, 128**  
   user-defined, **23, 30, 54**  
 Header command, **30, 54, 129**  
 Header width, **55, 108, 126, 128**  
   column, **136, 138**  
   row, **55, 108**  
 Hierarchy, mathematical, **71**  
 HP-75 functions, **78, 175-177**  
 HP 82162A Thermal Printer, **50, 116**  
 HP 82163 Video Interface, **50, 120**  
 HP 82905B Impact Printer, **50, 105, 110**

## I

---


/I. See *Insert command*  
 Inconsistent worksheet, **150**  
 Insert command, **64**  
 Installing a module, **16**  
 Integer format, **63, 133**  
 Internal Rate of Return, **152**  
☐ I/R (insert/replace) key, **44, 58**  
 IRR program, **45, 152, 178**

## L

---

Label, **29, 39, 44**  
 Label entry, **39, 41**  
 Language Extension (LEX) file, **166**  
 Left-Justified format, **133**  
 Left physical boundary, **27, 32**  
 Level  
   Command, **36, 46**  
   Edit, **38**  
   Top, **36, 37**  
 LEX file, **166**  
 List, **73**  
 Local column width, **136, 138**  
 Local format, **122, 132**  
 Local format field, **31**  
 Local Graphing format, **133, 140**  
 Logical operators, **70, 78**  
 Lower-right boundary, **109**

**M**

/M. See *Move command*  
 Magnetic card, 8, 20, 45, 103  
 Maintenance, 170  
 Manual recalculation, 80  
 Manual recalculation key () , 38, 58, 79, 80, 83, 84  
 Manual Recalculation mode, 80, 128  
 Mass storage device, 103, 104  
 Mathematical hierarchy, 71  
 MAXL (function), 45, 74, 176  
 MEAN (function), 45, 76, 176  
 Memory,  
   computer, 31, 47  
   multiple files in, 49  
 Memory management, 47  
 Message elements, 182-184  
 MINL (function), 45, 74, 176  
 Mixed coordinates, 28, 42  
 Mode,  
   Automatic Recalculation, 80  
   Coordinate Display, 62  
   Default Coordinate Display, 62  
   Formatted Display, 63  
   Formula Display, 21, 29, 42, 43, 62, 122  
   Manual Recalculation, 80, 128  
   Result Display, 21, 22, 29, 42, 43, 62  
   Unformatted Display, 63  
   User-Defined Coordinate Display, 62  
 Module,  
   installing, 16  
   removing, 17  
 Move command, 67  
 Moving  
   column or row, a, 67  
   cursor, the, 58

**N**

NA (function), 45, 77, 149, 153, 166, 176  
 Name field, 24  
 Notation, scientific, 71  
 Numeric expression, 70

**O**

Operand, 70  
 Operational levels, 36  
 Operator,  
   arithmetic, 70  
   logical, 70  
   relational, 70  
 Order of recalculation, 31, 80  
 Overflow display, 140

**P**

/P. See *Print command*  
 Parameter, 73  
 Parameter list, 73  
 Parentheses, use of, 38, 71, 73  
 Physical boundary, 27, 32  
 Precalculation, 79  
 Precision, 72  
 Print command, 106  
 Primary window, 34  
 Printer  
   boundary, 109  
   escape sequences, 110, 111  
   setup string, 110, 111  
   width, 110, 111, 112  
 Printer,  
   assigning the, 105  
   connecting the, 105  
   controlling the, 110  
   HP 82905B, 105  
   using another, 106  
 Printing  
   formulas, 106, 115  
   headers, 106  
   worksheet, 106  
 Print options menu, 106  
 Product information, 174  
 Prompt substitution, 182  
 Propagating NA and ERROR, 77  
 Purging worksheet file, 49

**R**

/R. See *Replicate command*  
 RAM, 47, 182  
 Random access memory (RAM), 47, 166, 182  
 Range, 73  
 Read-only memory (ROM), 166, 182  
 Recalculation, 79  
   automatic, 80  
   global, 80, 138  
   manual, 80  
   order of, 31, 80  
   problems with (circular or forward references), 83  
 Recalculation commands, 80  
 Recalculation mode field, 31  
 Recalculation order field, 31  
 Redefined keys, 61  
 Reference, external cell, 46, 143, 150  
 Relational operators, 70, 78, 79

Relative replication, **91**  
 Remaining memory field, **31, 49**  
 Removing module, **17**  
 Repeating Label command, **47, 61**  
 Replicate command, **85, 140**  
   both option, **93**  
   relative option, **91**  
 Replicating  
   formula, a, **91**  
   local format, a, **135**  
   range of cells, a, **96**  
   into a range, **98**  
   numbers and labels, **88**  
 Result Display mode, **21, 22, 29, 42, 43, 62**  
 Right-Justified format, **133**  
 ROM, **166, 182**  
 Rounding formula, **72**  
 Row,  
   deleting a, **66**  
   inserting a, **64**  
   moving a, **67**  
 Row header, **54**  
 Row header width, **55**  
**RTN** (return) key, **20, 25, 38, 40, 41, 42, 57, 58, 59, 60, 61, 63, 72, 86, 109, 111, 148**  
**RUN** (character left) key, **23, 29, 38, 62, 86, 96**  
 Running the VisiCalc program, **20**

## S

Scientific notation, **71**  
 Scrolling the cell display, **29**  
 Service  
   centers, European, **172**  
   centers, U.S., **172**  
   international, **173**  
   Repair charge, **173**  
   shipping instructions, **173**  
   warranty, **173**  
**SHIFT** **>** (Go To) keystroke, **22, 33, 38**  
**SHIFT** **APPT** (toggle cell display mode) keystroke, **37, 63, 134, 140**  
**SHIFT** **ATTN** (exit VisiCalc) keystroke, **16, 24, 25, 27, 37, 47, 58, 170**  
**SHIFT** **DEL** (shift delete) keystroke, **58**

**SHIFT** **EDIT** (toggle status line) keystroke, **30, 31, 37**  
**SHIFT** **FET** (define access cell) keystroke, **34, 37**  
**SHIFT** **I/R** (shift insert/replace) keystroke, **58**  
**SHIFT** **RUN** (character right) keystroke, **23, 29, 38, 62, 86, 96**  
**SHIFT** **TAB** (window width right) keystroke, **29, 38, 62, 86, 96**  
**SHIFT** **TIME** (toggle cell contents) keystroke, **37, 62, 63, 96, 116**  
 Slash key, **[/]**, **38, 46**  
 Source range, **86**  
 STANDBY setting, **20**  
 Start-up, **20, 24**  
 Status line, **30**  
 Suppressing headers, **55, 108, 126, 128**  
**SUM** (function), **23, 45, 73, 176**  
 System requirements, **50**

## T

**TAB** (window width left) key, **29, 38, 62, 86, 96**  
 Target range, **86**  
 Technical assistance, **174**  
 Terminating an entry, **38, 40, 41, 42, 58, 61**  
**TIME** (toggle cell coordinates) key, **23, 37, 58, 62, 96, 109, 116**  
 Toggle, **21**  
 Top level, **36, 37**  
 Top level, returning to, **41, 47**  
 Top physical boundary, **27, 32**  
 Trigonometric functions, **175, 177**

## U

Underlined characters, **185**  
 Underlined colon, **21, 29**  
 Unformatted Display mode, **63**  
 Upper-left boundary, **109**  
 Use of brackets, **30, 38, 40, 54, 61**  
 Use of caret (^), **30, 38, 70, 71**  
 Use of parentheses, **38, 71, 73**  
 User-Defined Coordinate Display mode, **62**  
 User-defined coordinates, **28, 30, 42**  
 User-defined headers, **23, 30, 54**

## V

---

*/V. See Video command*  
 Value, 29, **41**, 44  
 Value entry, **41**, 44  
 Video command, **47**, **121**  
 Video display,  
   activating the, **120**  
   assigning the, **119**  
   connecting the, **119**  
   deactivating the, **120**  
   HP 82163, 50, **120**  
   interpreting the, **123**  
   using the, **121**  
 Video display  
   device, 119, **120**  
   length, **186**  
   parameters, **186**  
   width, **186-187**  
 Video interface, 119, **120**  
 VisiCalc  
   built-in functions, 72, **175-177**  
   card pac, **8**  
   display, **27**  
   mode, **19**  
   operational levels, **36**  
   start-up message, **20**, 24  
 VisiCalc, executing, **20**, 24  
 VISIDATA, **155-158**, 159, 160, 165  
 VISIMSGS, **182**, **185**

## W

---

*/W. See Width command*  
 Warranty, **170**  
 Warranty information, **171**  
 Width,  
   global column, **138**, **139**  
   header, **55**, 108, 126  
   local column, **136**, 138  
 Width command, **47**, 122, 128, **136**, 138, 139  
 Window, **34**  
   alternate viewing, **34**, 35  
   primary, **34**  
 Window access cell, **34**  
 Working message, **20**, 128  
 Worksheet,  
   accessing another, 46, **143**  
   active, **27**  
   creating a, 20, 24, **25**  
   formatting the, **131-140**  
   inconsistent, **150**  
   simplifying a, **144-146**  
 Worksheet catalog, **25**  
 Worksheet file,  
   creating a, **25**  
   purging a, **49**  
   retrieving a, **25**  
   storing a, **103**  
 Worksheet file name, **24**, 25, **26**  
 Worksheet name  
   field, **31**  
   template, **20**, **24**  
 Worksheet size, **47**  
 Worksheet status line, **30**





**How To Use This Manual (page 7)**

- 1: Getting Started (page 15)**
- 2: Creating and Editing Worksheets (page 53)**
- 3: Using Functions and Formulas (page 69)**
- 4: Using the Replicate Command (page 85)**
- 5: Storing and Retrieving Worksheet Files (page 103)**
- 6: Using a Printer with VisiCalc® (page 105)**
- 7: Using a Video Display with VisiCalc® (page 119)**
- 8: Formatting the Worksheet (page 131)**
- 9: Accessing Another Worksheet From a Cell (page 143)**
- 10: Extending the VisiCalc® Function Set (page 151)**
  
- A: Owner's Information (page 169)**
- B: VisiCalc® Functions (page 175)**
- C: The IRR Extension Function (page 178)**
- D: Changing to Foreign Language Prompts (page 181)**
- E: Setting the Video Display Parameters (page 186)**
- F: Glossary (page 188)**

**Command and Operation Summary and Index (page 201)**

**Subject Index (page 211)**



Portable Computer Division  
1000 N.E. Circle Blvd., Corvallis, OR 97330, U.S.A.