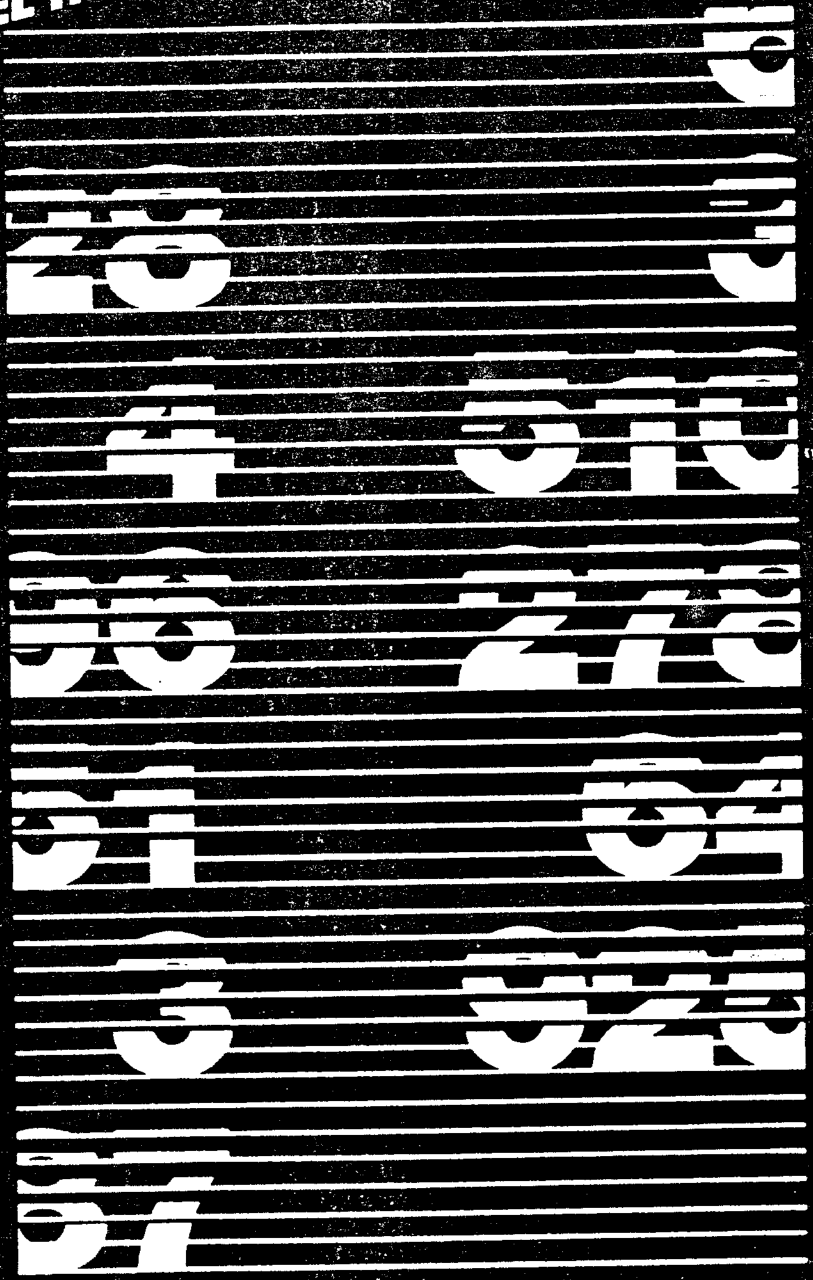
ERSONAL SOFTWARE™

VISICALC®

TRS-80® MODEL III USER'S GUIDE



Radio Shack®

TRS-80

MODEL III
MICRO
COMPUTER

Catalog Number 26-1569

TRSDOS 1.3 VisiCalc® Update
TRS-80 Model III

IMPORTANT INFORMATION

The TRS-80 Model III VisiCalc® program diskette in this package contains TRSDOS version 1.3. If you have VisiCalc data files created with an earlier TRSDOS version (1.2) of the VisiCalc program, and want to use them with the current version, the following instructions apply to you. You can determine the TRSDOS version number on any diskette containing TRSDOS by placing the diskette in drive 0 and turning on the computer (or pressing the RESET button if the computer is already on). The TRSDOS version number will be displayed on the screen along with the copyright information.

If you try to write a file on a storage diskette in drive 0, and that diskette contains an earlier version of TRSDOS, the program will freeze and the worksheet you are trying to save will be lost. *Therefore, upgrade all VisiCalc data and system diskettes to TRSDOS 1.3 data or system diskettes before using them with your new VisiCalc program.*

Make sure you have backup copies of all your storage diskettes. You may want to use them again with a program that only uses an earlier version of TRSDOS. Once the files have been upgraded to TRSDOS 1.3 diskettes, they should not be used with older versions of the VisiCalc program.

If you have a single-drive TRS-80 Model III, you will not be able to upgrade your old storage diskettes yourself. You must have a two-drive system to run the XFERSYS upgrade program. See your Radio Shack Computer Center or Computer Marketing Representative for assistance in upgrading your diskettes.

If you have a two-drive TRS-80 Model III, use the XFERSYS utility which is on the TRSDOS 1.3 system diskette to upgrade storage diskettes to TRSDOS 1.3 storage diskettes. The write protection on the VisiCalc program diskette will not allow it to be upgraded. If you XFERSYS a VisiCalc program diskette, you will upgrade the data files on that diskette but the VisiCalc program itself will be unusable. Therefore, COPY data files contained on your program diskette to another storage diskette (FORMATted with the earlier version of TRSDOS) and then run the XFERSYS utility, using the TRSDOS 1.3 system diskette and the storage diskette containing the copied files.

1. Insert a system diskette containing the earlier version of TRSDOS into drive 0 and press RESET. Use the DIR command to check that there are at least 13 free granules on the old storage diskette. If there are not, COPY some of the files to another diskette FORMATted with this earlier version of TRSDOS and DELETE them from the original diskette. (XFERSYS will be performed upon both storage diskettes.)

2. Make backup copies of all the storage diskettes (for safekeeping). Use the BACKUP command which is on the system diskette currently in drive 0. Label the backup copies with the TRSDOS version number and put them away.

3. Now insert a TRSDOS 1.3 system diskette into drive 0 and press RESET. Put the old TRSDOS storage diskette in drive 1. Both the source and destination diskettes must be write-enabled. Run the XFERSYS program to upgrade the diskette in drive 1. If you started with an old system or data diskette, the XFERSYSed diskette will be a new system or data diskette, respectively, for use with the TRSDOS 1.3 version of the VisiCalc program. Be sure to write the new TRSDOS version number on the diskette label.

See your TRS-80 Model III *Disk System Owner's Manual* for more information on XFERSYS, BACKUP, COPY, and DELETE.

Table of Contents

PART I. INTRODUCTION

PART II. VISICALC TUTORIAL

Lesson One

Lesson Two

Lesson Three

Lesson Four

PART III. VISICALC COMMAND REFERENCE

PART IV. INDEX

SUPPLEMENT: PROGRAMMER'S GUIDE to DIF™

Software Arts Technical Note SATN-18/TRS-80
Special Edition for the Radio Shack TRS-80 Model III

(reprinted with permission of Software Arts, Inc.)

Table of Contents

PART I. INTRODUCTION

How To Use this Manual

This manual is divided into four parts. It has been designed with the consideration that different people using the VisiCalc® program will have differing levels of computer experience.

Part I contains an overview of the VisiCalc program, information about your equipment needs, and instructions for loading the VisiCalc program and making blank diskettes ready for use.

Part II is a step-by-step Tutorial in the use of the VisiCalc program with your TRS-80 Model III. Those with little or no experience with personal computers will find that it anticipates many of the questions and problems that may arise. The Tutorial comprises four lessons which guide you from the point at which you finish loading the VisiCalc program in Part I, through several examples that show you how to use the program and your computer to solve problems in your professional work and your everyday life. Each lesson will show you exactly what to type keystroke by keystroke, and should be done with the computer in front of you. As you practice, you'll gain familiarity and confidence in using some of the more advanced features of the VisiCalc program. Before long, you'll need only Part III and the VisiCalc® Pocket Reference.

Part III is the VisiCalc Command Reference. It contains a chart of VisiCalc commands illustrating their relationship, notes on the elements of the VisiCalc screen display, and a detailed discussion of each command with examples of its use. You will probably find yourself referring to this section frequently, especially as you use the advanced features of the VisiCalc program to speed your work and do complicated applications. The commands presented in this section of the manual are summarized on the VisiCalc Pocket Reference which you'll find in the pocket on the inside back cover of the binder holding this manual.

There is an Index for this manual, listing subjects alphabetically, following Part III.

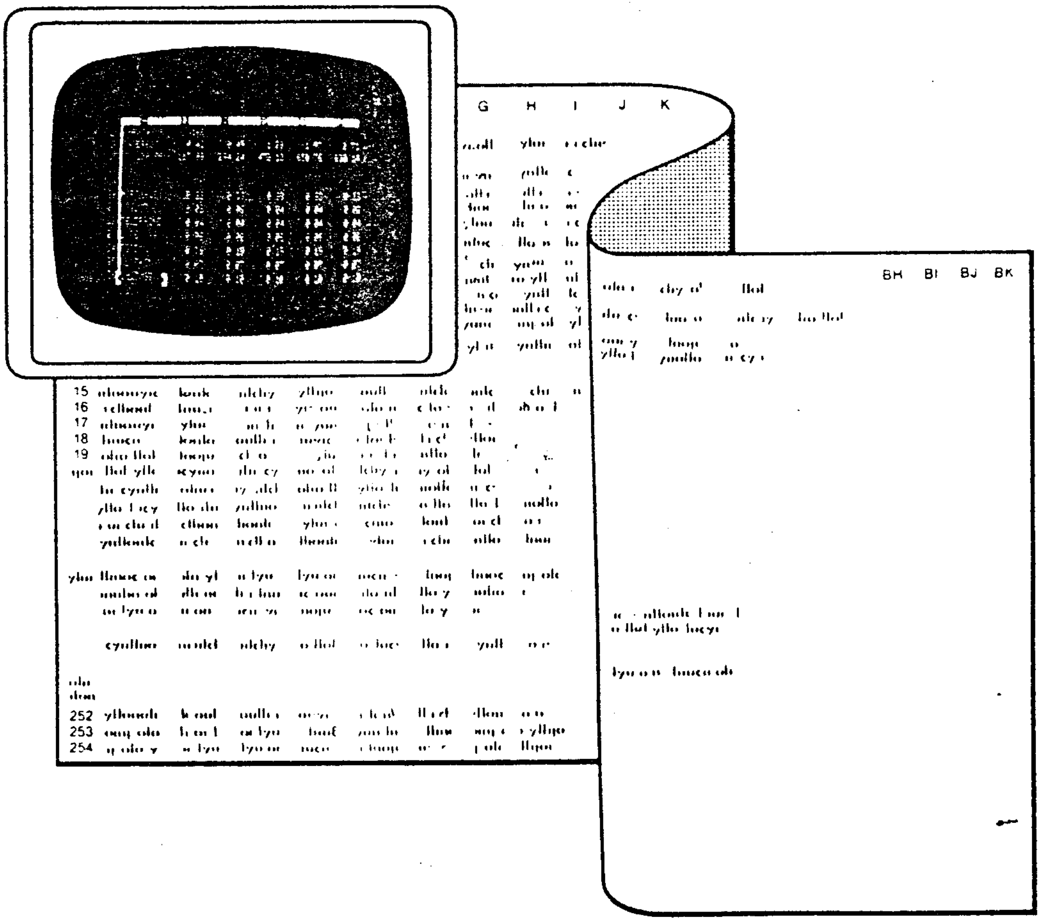
The Supplement is the Programmer's Guide to DIF™.

The best way to learn to use the VisiCalc program is to try it. Don't be afraid to experiment. Trying out ideas will help you answer a lot of your questions, while enhancing your experience and confidence. Entering "odd" data and patterns will hurt neither the computer nor the VisiCalc program.

Overview of the VisiCalc Program: the "Electronic Sheet"

The VisiCalc program was born out 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, and even balancing your checkbook, is done with a calculator, pencil and paper.

The VisiCalc program combines the convenience and familiarity of a pocket calculator with the powerful memory and electronic screen capabilities of the personal computer. With the VisiCalc program, the computer's screen becomes a "window" which looks out upon a much larger "electronic sheet." You can move, or "scroll," this window in all four directions to look at any part of this sheet, or you can split the computer screen into two "windows" to see any two parts of the sheet at the same time.



YOUR SCREEN IS A WINDOW INTO THE ELECTRONIC SHEET IN THE COMPUTER'S MEMORY

The sheet is organized as a grid of columns and rows. The intersecting lines of the columns and rows define thousands of entry positions. At each position you can enter an alphabetic title, a number or a formula to be calculated. Just by "writing" on the sheet, you can set up your own charts, tables and records. Formatting commands let you individualize the appearance of each entry, row or column. If you wish, for example, you can make your VisiCalc checkbook record look just like your bank statement.

The power of the VisiCalc software is that your computer *remembers* the formulas and calculations you use as you work through a problem. If you change a number you had previously written on the electronic sheet, all other related numbers on the sheet change before your eyes, as the VisiCalc program automatically recalculates all of the relevant formulas.

Recalculation makes the VisiCalc program 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 the VisiCalc program. You may want to know what the impact on your company will be if a specific product doesn't sell as well as you anticipate. What if you sell only 200 "widgets" a month instead of 250? What if you sell 300? What if one of your salesmen quits and it takes his replacement six weeks to come up to speed? Playing "what if" with the VisiCalc program is usually a matter of changing a single number. Doing the same thing with a calculator, pencil and paper might take hours of erasing and recalculating.

The VisiCalc editing features let you change, insert or delete titles, numbers, or formulas. The existing VisiCalc chart or table is instantly restructured, with all of the columns, rows, and other formulas edited to reflect your changes.

If you've entered a formula at one position, the VisiCalc program lets you replicate it at any number of other positions. The VisiCalc program will also add up, average, or otherwise manipulate rows, columns, or other ranges of numbers.

Once you've established the format for a particular application, you just enter or change numbers. You can save the entire electronic sheet on your diskette, and you can print all or part of the sheet on a printer.

You can learn the elementary features of the VisiCalc program in an hour or two, and you'll find that you are immediately able to solve simple problems. As you use the VisiCalc program for more complicated applications, you'll discover that it has a broad range of features and commands. You can learn these features and commands as the need arises.

What You Need

To use the VisiCalc program, you will need the following:

1. The TRS-80 Model III Desktop Business computer which contains Model III BASIC, 48K of RAM memory and two disk drives. Minimum requirements are a TRS-80 Model III with Model III BASIC, 48K of RAM memory and one disk drive.
2. The VisiCalc program diskette. There are two program diskettes enclosed in the inside front cover of the binder holding this manual.
3. At least two blank 5" floppy diskettes. This will be enough to get you started, but you will eventually need more and should plan for it.
4. If you want printed copies of your VisiCalc electronic sheets, you need any TRS-80 Printer.

Setting Up Your Equipment

Set up all the equipment according to the instructions in the manuals that accompany the Model III.

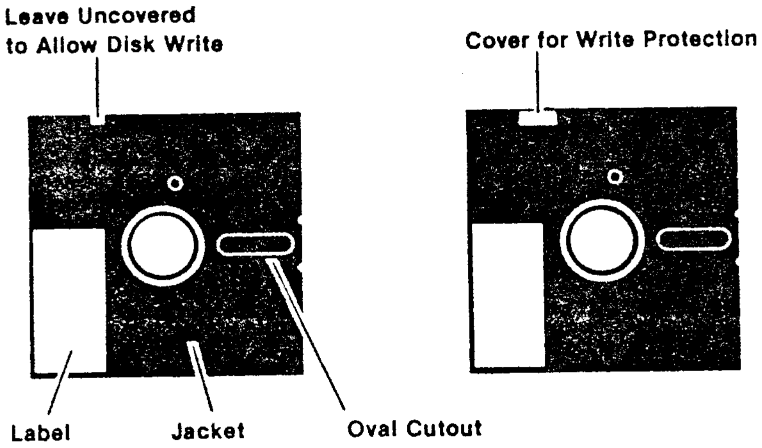
The TRS-80 Model III computer drives are referenced in this manual as follows: the bottom drive is Drive 0 (zero), and the top drive is Drive 1.

Your VisiCalc program package contains two program diskettes. Use one of these as your working copy and put the other away in a safe place as a backup. If your working copy is no longer usable for whatever reason, use your backup VisiCalc program diskette as your working copy. Contact your Radio Shack store or Computer Center for instructions on how to exchange the unusable VisiCalc program diskette for another. You will be required to present the diskette to your dealer. When you receive the replacement, put it away as the backup program diskette.

A Word on Diskettes

A word of caution about the handling of diskettes—you can't be too careful with them. Each diskette is a magnetically coated plastic disk, inside a protective square plastic cover. This cover has an oval cutout in it. Through the oval cutout in the square cover, you can see the magnetic surface of the actual diskette. *Never* touch the exposed magnetic surface with your fingers or any implement.

Sometimes there is a small square notch cut out of one side. The square notch is sensed by the disk drive and tells the computer that it may record information on the diskette. To record on a storage diskette, the notch must be uncovered.


Leave Uncovered to Allow Disk Write
Cover for Write Protection
Label
Jacket
Oval Cutout

Protect the diskette from dust by storing it in the paper sleeve that it comes in. Keep it at least 6 inches from magnetic fields such as those generated by a TV. Extremes of temperature (such as a car trunk on a warm day) could destroy a diskette, and you would lose your data, or your VisiCalc program. Don't bend or staple the diskette, or write on the square plastic cover with a hard pen or pencil (use only the soft felt tip pens). Always handle diskettes gently, keeping them away from magnetic fields, dirt, and liquids.

Drive 0 makes an audible sound when it is running and the red ACTIVE light comes on. Never open the disk drive door or insert or remove a diskette while the drive is running; this will probably damage the diskette and may also damage the drive. Never turn the power to your computer off or on while a diskette is in the drive.

Getting Started

You will need the two blank 5" diskettes listed under "What You Need" above (you may use previously used diskettes which you no longer want) and your working copy of the VisiCalc program diskette. Before you use the VisiCalc program the first time, you must:

1. Get TRSDOS started. TRSDOS stands for the TRS-80 Disk Operating System and contains the programs which run your computer and disk drives.
2. Use the TRSDOS BACKUP command to make some VisiCalc storage diskettes.

Loading the TRSDOS Programs

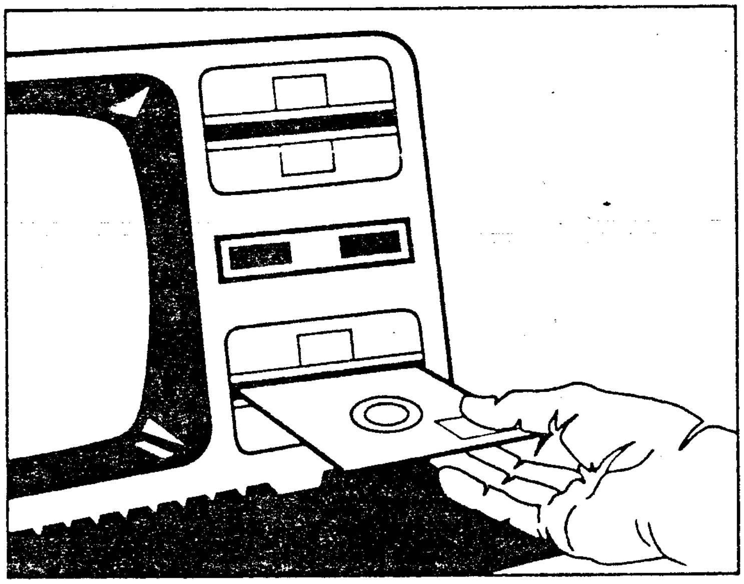
Your VisiCalc program diskette also contains the TRSDOS programs. TRSDOS is a short way to say TRS-80 Disk Operating System. These programs get your computer started before you can start the VisiCalc program. You must use the TRSDOS on the VisiCalc diskette to run VisiCalc and make storage diskettes.

1. Turn all peripherals on.

2. Turn on your computer. Drive 0 will whir and the ACTIVE LIGHT will come on.

   Wait until all disk drives stop and the lights are out.

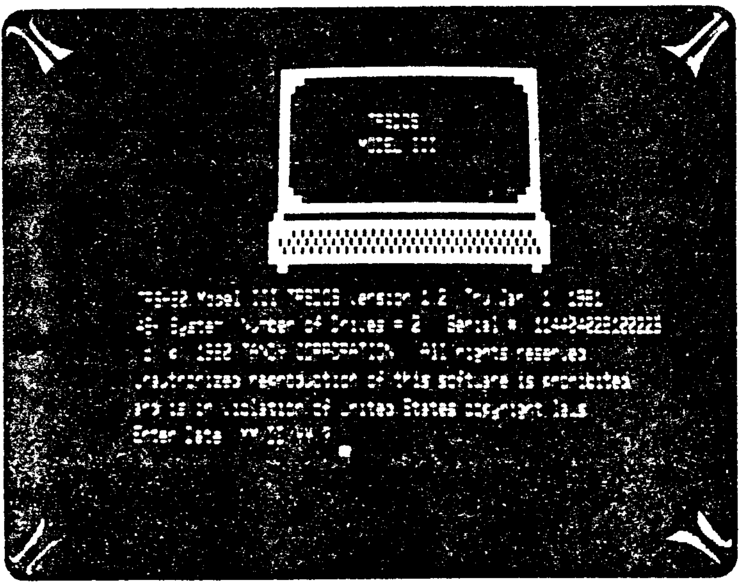
3. Open the door on Drive 0 by gently pulling on the lower edge. Insert the VisiCalc diskette in Drive 0 with the label facing up, the oval cutout in the square diskette jacket entering first. Gently push the diskette all the way in, close the door, and press the RESET button.



The drive will run for about half a minute and some messages will appear on the screen. If you see the message

```
NOT A SYSTEM DISK
```

you do not have the VisiCalc program diskette in drive 0. If you have the VisiCalc program diskette in drive 0, then after a few seconds your screen should resemble the photo below.



The TRSDOS version number and date of creation will be displayed, followed by the amount of RAM and the number of drives in the system. At the end of the copyright statement you will see

```
Enter Date (MM/DD/YY)?
```

4. Type the date in the exact format requested. For example, you would type **07/04/80** for July 4, 1980. Then press the **ENTER** key. TRSDOS will not continue until you have entered the date correctly. A prompt for the time will appear.

```
Enter Time (HH:MM:SS)?
```

5. TRSDOS uses a 24 hour format so you would enter **15:32:00** for 3:32 p.m. If you don't want to set the time, just press **ENTER** and TRSDOS will set the time to 00:00:00.

TRSDOS will now display the message

```
TRSDOS Ready
```

You can now use TRSDOS commands to backup and initialize storage diskettes before you begin using the VisiCalc program.

If your computer does something different at any point in the above instructions, remove the VisiCalc diskette from DRIVE 0, turn off your computer, wait 15 seconds, and try again.

You now need to make at least two storage diskettes for saving VisiCalc worksheets.

Initializing VisiCalc Storage Diskettes

Since most of the electronic sheets you develop with the VisiCalc program will contain valuable data which you'll want to keep, you must make some storage diskettes. The blank diskettes.that are listed in the section "What You Need" to use with the VisiCalc program must be organized with special instructions written on them by the TRS-80 Model III TRSDOS (Disk Operating System) before you can actually use them to store your data. This is the initialization process.

DO NOT use any version of TRSDOS other than the version on your VisiCalc program diskette for initializing or making backups of VisiCalc storage diskettes.

To initialize storage diskettes for use with VisiCalc, follow the procedures in "Instructions for Using BACKUP" for making a copy using BACKUP with the VisiCalc diskette. Put a distinctive label on each diskette such as VISICALC STORAGE DISKETTE 1 and VISICALC STORAGE DISKETTE 2. It is also a good idea to write the date on each diskette label. Apply these labels to the diskette jackets on their manufacturer's label. If the title label is already on the diskette, be sure to write with a felt tip pen, not a ballpoint.

If you have more than one drive, you can use diskettes formatted with the TRSDOS FORMAT Command as storage diskettes (see your Disk Operating System Manual). Remember you must always have a diskette with TRSDOS on it in Drive 0. Diskettes formatted with the FORMAT Command can only be used in a drive other than Drive 0.

In the initialization process, the computer is recording a pattern on the surface of the diskette, so that the VisiCalc program can find a given spot on the diskette surface and "write" information there or "read" it back later. What's more, information "written" by one program (such as VisiCalc) can be located later and "read" by a different program (which could be written in BASIC). (See the Supplement.) If you initialize a diskette that has had data stored on it from some previous use, either by VisiCalc or some other program, that data will be erased by this process.

You must have a storage diskette already initialized before you try to save the data from a VisiCalc electronic sheet. You should always have at least one extra initialized storage diskette on hand. Having this extra storage diskette will help insure that you don't lose data because a diskette is full and the VisiCalc program cannot save a sheet on it.

Instructions for Using BACKUP

If you have followed the procedures in "Loading the TRSDOS Programs" your screen should now show:

```
TRSDOS Ready
```

Make sure the write protect notch on all diskettes, except the VisiCalc program diskette, is *uncovered* when you use BACKUP. Put the VisiCalc program diskette in Drive 0 and close the door.

Next type the word

```
BACKUP
```

and press ENTER. The words

```
TRSDOS Model III Backup Utility Ver x.x
```

will appear on the screen followed by

```
SOURCE Drive Number?
```

Type **0** (zero) which is the number of the source drive and press **ENTER**. The next prompt is

```
DESTINATION Drive Number?
```

If you have only one drive type 0. If you have more than one drive, type the number of the drive containing the diskette you will backup onto.

If you have only one drive, you will be prompted when to insert the source diskette and the destination diskette. **Make sure your diskettes are clearly labeled and that you are inserting the correct diskette.** Be sure the diskette you wish to backup onto has the write protect notch uncovered, is in the drive, and the door is closed. Press **ENTER**. You will see the prompt

```
SOURCE Disk Master Password?
```

The VisiCalc disk is your source diskette and its password is: PASSWORD. Type **PASSWORD** and press **ENTER**.

This is the next message that will appear on the screen.

```
Source Drive        Destination Drive
```

with the numbers of the drives you have selected. Make sure the source diskette and the destination diskette are in the correct drives.

The next two prompts will appear only if you are backing up onto a diskette which previously contained data. You will be told

```
Diskette contains DATA. Use Disk or not?
```

You may answer **Y** (Yes) or **N** (No). (If you answer N, the prompt `TRSDOS Ready` will appear on the screen and the BACKUP procedure will be terminated.) Press **Y** to continue and you will see

```
Do you wish to RE-FORMAT the diskette?
```

(If you press N, BACKUP will use the format that is already on the diskette and skip the formatting and check of flawed tracks.) Press **Y** to continue, and the BACKUP utility program will run.

It takes a little over 2 minutes for the computer to format and backup a diskette. Messages will appear on the screen as it formats and verifies the tracks on the destination diskette. The message `xx Flawed Tracks` will appear (you don't need to be concerned about any flawed tracks). TRSDOS then reads the data from the source diskette and writes and verifies it on the destination diskette. This is explained more fully in your TRSDOS Manual.

If your BACKUP was successful you will see:

```
* * Backup Complete * *
```

followed by

```
TRSDOS Ready
```

If you don't get this message, read the instructions in your TRSDOS Manual again carefully, and try BACKUP again.

Repeat these BACKUP procedures on your second blank diskette. When you have finished you will have two initialized VisiCalc storage diskettes containing TRSDOS. You will load and save VisiCalc worksheets to and from these diskettes.

Now you are ready to load the VisiCalc program into the computer's memory.

Loading the VisiCalc Program

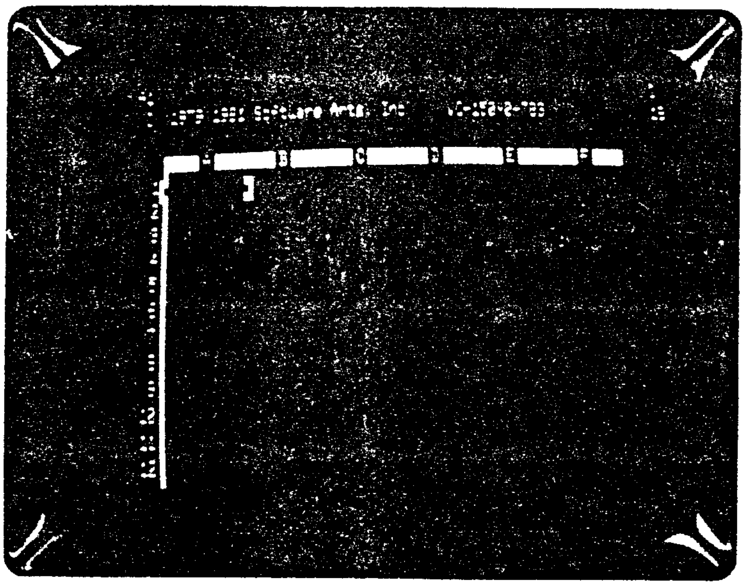
Make sure that your working copy of the VisiCalc program is in DRIVE 0 and that TRSDOS Ready is on the screen.

Then type

```
VC
```

and press **ENTER.**

The VisiCalc screen will appear, looking like the photo below, and you are ready to begin using the VisiCalc program.



To minimize wear on your working copy of the VisiCalc diskette, remove it from the drive and put it away. You will not need it again until the next time you turn on your computer and load the VisiCalc program. Insert a storage diskette containing TRSDOS in DRIVE 0 and you are ready to save VisiCalc worksheets.

If your screen doesn't resemble the photo, you may have made a mistake or there may be a problem with your hardware. Remove your VisiCalc diskette from DRIVE 0. Then turn off your computer and try again, using the "Loading TRSDOS" and "Loading the VisiCalc program" instructions. Remember to wait 15 seconds before turning your computer back on again. If after several tries you are not successful at loading the VisiCalc program, see your Radio Shack Computer Center for help.

Sometimes your TRS-80 Model III gives you helpful messages. If you see the message:

```
NOT A SYSTEM DISK
```

or

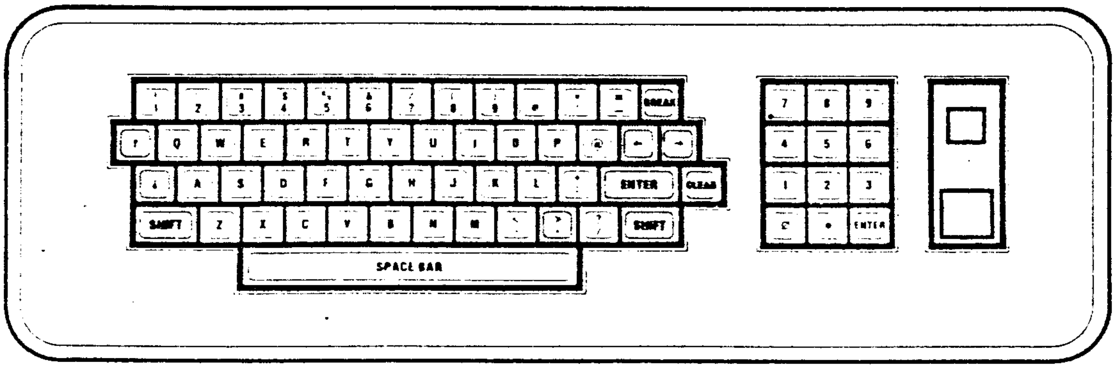
```
PROGRAM NOT FOUND
```

you did not have the VisiCalc program diskette in drive 0. Find it, reinsert it correctly, and then press the **RESET** button on the front of your computer. Enter the date and time again and when the message TRSDOS READY appears on the screen, type **VC** in capital letters and press **ENTER**. If some other message appears, you can look it up in the section entitled "ERROR MESSAGES" in your *TRS-80 Model III Owner's Manual,* or type ERROR nn (nn is the number of error) and the message will appear.

Some Notes on Your Keyboard

Your keyboard has a number pad built in at the right of the regular keyboard. The numbers on it may be used interchangeably with the numbers on the top row of the regular keyboard.

Your Model III computer types in both upper and lower case letters. In the VisiCalc program, you will automatically get lowercase letters except in commands. To use upper case, hold down the SHIFT key while you are typing a letter—there is no shift lock. VisiCalc will change some letters to upper case automatically if it is necessary. For example, all file names and all commands will be in upper case even if you type them in lower case.



Several characters that are used repeatedly in the VisiCalc program must be typed with the **SHIFT** key depressed as you would on a typewriter. Except for an occasional reminder, we do not instruct you to hold down the **SHIFT** key to type the characters. Also note the positions of the keys that are shaded on the keyboard. You will use them a great deal. You will see the key labeled **ENTER** represented by this symbol, Ⓔ, throughout this manual.

The keys with arrows on them at the right side of the keyboard are used to move the highlight which you will see on your VisiCalc sheet. The following are the special symbols we use in this manual and the keystrokes they refer to:

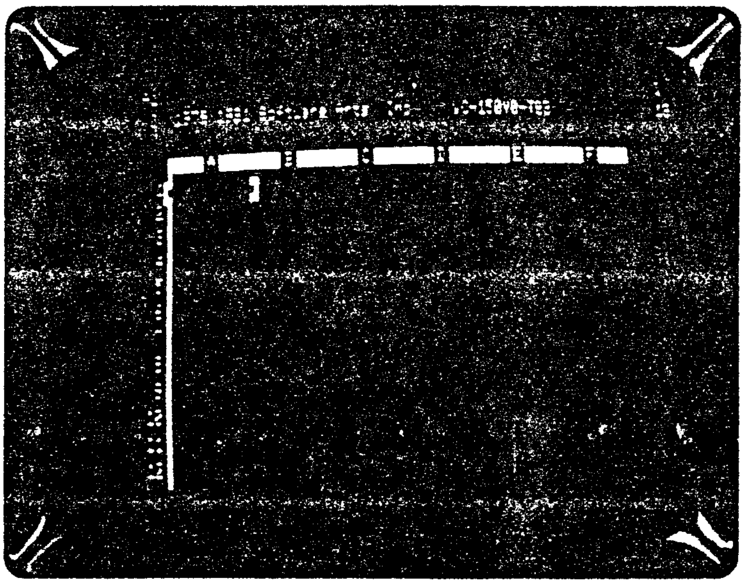
Symbol	Keystrokes
Ⓔ	Press ENTER
▼	Press ↓
▲	Press ↑
▶	Press →
◀	Press ←
∧	Hold down SHIFT and press @

PART II. VISICALC TUTORIAL

Lesson One

Lesson One

When you have loaded the VisiCalc program into your computer, as described in the section entitled "Loading the VisiCalc program," the image on your screen should resemble the photo shown in that section. The same photo is reproduced below. If your screen is different, type the following keys: /CY This will clear the sheet and it will look like the photograph. Here we'll examine the components of this screen image more closely.



Your screen has become a **window** into the computer's memory, which the VisiCalc program has organized like an **electronic sheet.** As you can see, the sheet is divided into rows which are numbered 1, 2, 3, and so on, and columns which are lettered A, B, C, and so on. At each intersection of a row and column there is an **entry position,** with a **coordinate** such as A1, B3, C17, and so forth. At each entry position you can "write" a message or title, a number, or a formula of the kind you might enter, keystroke by keystroke, on a calculator. In a moment we'll demonstrate how you move around and write on this electronic sheet.

Above the white border with the column letters, there are three additional lines which make up VisiCalc's **control panel.** The middle line of this control panel displays the VisiCalc copyright notice followed by the **version number.**

Should you ever need to call or write to Radio Shack to ask questions about the VisiCalc program or to report problems, please be sure to include this version number and the model of your computer.

Press the key marked ENTER. As we mentioned in the section "Notes on Your Keyboard" in Part I, we'll indicate the ENTER key with the symbol Ⓔ. The copyright notice and version number will disappear when you press Ⓔ or any other key. Now press the / key followed by the V key and both will reappear. Just type /V anytime you want to see your version number.

Moving the Cursor

Look at the point where column A and row 1 intersect. This is coordinate A1. To keep you from getting lost on the electronic sheet, the VisiCalc program prints the coordinate for you on the upper line of the top border of the sheet. Notice that there is a pair of brackets over the entry position at column A, row 1. These highlight the entry position and are called the **cursor** or **highlight**. You always write on the electronic sheet at the position marked by this cursor; you can think of it as the point of your pencil or pen. You move the cursor with the four arrow keys located on your keyboard. The right and left arrow keys are on the right side and the up and down arrow keys are on the left. These keys control the direction of movement of the highlight. Throughout this manual, the symbols ⬌ and ⬍ will mean to move the highlight down or to the right with these keys.

Try pressing the right arrow key ⬍ once. Notice that the cursor moves to the position at column B, row 1, and the cursor coordinate in the upper left corner of the control panel changes to B1. (The copyright notice and version number will also disappear with your first keystroke.) Now press the left arrow key ⬍ and watch the cursor move back to its original position. Try moving the cursor down to row 2 of column A with ⬌ and then back to position A1 with ⬌.

If while you are practicing, you mis-key an entry and see either the word "Value" or "Label" appear on the second line of the control panel, don't worry. Just press the key in the upper right corner of the keyboard, marked BREAK, and the word will disappear (and the number or letter beneath it). We'll go into more detail on all this a little later in the manual.

Scrolling the Window

When you first load the VisiCalc program, your screen window is positioned to let you look at the upper left hand corner of the VisiCalc electronic sheet. The window allows you to see the first six columns (A through F) and the first twelve rows (1 through 12) of the sheet. Now, with the cursor at A1, press the right arrow key ⬍ several times until the highlight is at the right edge of the window. Now press the right arrow key ⬍ again. Notice that the next column to the right comes into view, while column A disappears off the left edge of the window. When this happens, we say that the window has **scrolled** to the right. Try pressing ⬍ a few more times, watching more columns appear to the right edge of the window, and disappear at the left.

The screen window will also scroll to the left. (In fact, it will scroll in all four directions.) Press the left arrow key ← until the cursor is at the left edge of the window. Then press ← several more times and notice that the columns that had disappeared as you scrolled the window to the right come back into view. Press ← until the cursor is back at position A1. Now try pressing ← one more time. You should see the coordinate on the top line of the screen flash. This is the program's way of telling you that you are bumping into the edge of the sheet.

Press the up-pointing arrow key ↑ which would normally move the cursor up. The cursor is bumping into the edge of the sheet again. So far we have encountered the left edge and the top edge of the sheet. Now, we will go looking for the other two edges.

Press the down arrow key ↓ until the cursor has moved down to position A12. Then press ↓ again, and notice that row 13 comes into view, while row 1 disappears off the top of the screen window. Continue pressing the ↓ key five or six more times. As you can see, the electronic sheet is quite a bit larger than an ordinary sheet of paper.

Automatic Repeat

We can speed our search for the bottom edge of the sheet by using the automatic repeat feature of the computer. To see this in action, press the ↓ key again, and this time **hold it down.** (The cursor and the window should be scrolling downwards automatically.) Continue to hold down the ↓ key until you bump into the bottom edge of the electronic sheet. The cursor will be at position A254.

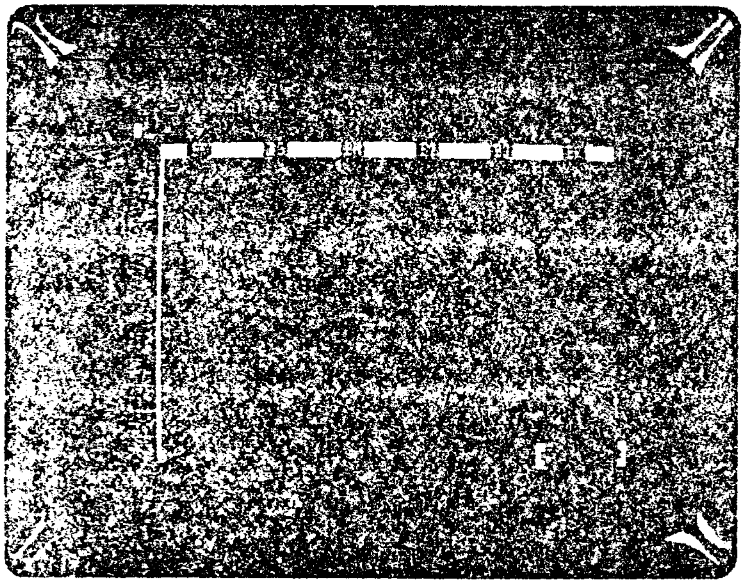
Now, let's search for the right hand edge of the sheet. Press → and hold it down. The cursor and window will go scrolling off to the right. As the cursor and window scroll to the right, notice how succeeding columns are lettered. After A, B, C, . . . , X, Y, Z comes AA, AB, AC, . . . , AX, AY, AZ, and then BA, BB, BC, ... The cursor finally stops at position BK254, as it bumps into the right hand edge of the sheet. You are now at the lower right hand corner of the VisiCalc electronic sheet.

Direct Cursor Movement

Even with the aid of automatic repeat, it took a while to scroll the cursor and window all the way to the lower right corner of the VisiCalc sheet. There's an easy way to move an arbitrary distance across the sheet with a few keystrokes.

Type the character >. If you make a mistake, press the CLEAR key to correct what you've typed. The next section "Backing Up the Edit Cursor" explains the use of this key in detail.

Two things will happen: 1. The message Go to: Coordinate appears on the middle line of the control panel, at the top of the screen. 2. Directly below this message, a box or edit cursor appears. You have discovered two new components of the VisiCalc control panel: the **prompt line** and the **edit line.**



The VisiCalc program communicates with you on a keystroke by keystroke basis, just like a pocket calculator. Each time you press a key, the VisiCalc program tells you, on the prompt line, what you can type next. Right now, the prompt line is telling you that the program has recognized your keystroke command >, which means **GO TO** an arbitrary position on the sheet, and that next, the VisiCalc program expects you to type the COORDINATE of the position to which you want the highlight cursor to move.

Press the **A** key once: The letter A will appear on the edit line (third line from the top of the screen), followed by the edit cursor. Notice that you get a capital A, whether you press the SHIFT key or not. The VisiCalc program knows that you are entering a coordinate and so takes care of upper case for you. The SHIFT key matters only when you press a key with two symbols on it, or as you will see, when you are entering labels. Now press the number **1** key: we want to move the highlight cursor back to position A1. You may use either the number on the keyboard or the number on the number pad. Do not use lower case "L" or upper case "I" instead of the 1 key or the letter "O" for a zero. So far, we have A1 on the edit line, highlighted by the edit cursor. VisiCalc is still waiting for you to type something: it doesn't know yet whether you want to go to position A1, or position A11, or A121, or some other position. Now press the Ⓔ key. The information on the prompt and edit lines disappears, and the cursor moves back to the upper left corner of the sheet.

Try another example. Press the keys **>C10**Ⓔ Does the highlight cursor move to the expected position?

Backing Up the Edit Cursor

The VisiCalc program has several error correction features, each of which will be covered in this lesson. The first of these is the key labeled CLEAR.

Press the following keys: **>A11** Then pause for a moment before pressing Ⓔ Suppose that you intended to move the cursor to position A1, but you accidentally pressed the 1 key twice. You now have A11 on the edit line, followed by the edit cursor.

Press the key marked CLEAR once. Notice that the edit cursor "backs up" one character and erases the extra 1, leaving you with A1. Now press Ⓔ The highlight cursor will move back to the upper left hand corner of the sheet. In general, the VisiCalc program will let you correct typing errors by backing up with the CLEAR key. You can back up more than one character. For example, to change A11 to A2, you would press the CLEAR key twice, backing up to leave just the letter A, and then you would press the **2** key to get A2.

Besides backing up, you can "back out" with the CLEAR key. Press the following keys: **>B5** Then pause. Suppose that you change your mind and decide that you don't want to move the cursor at all. Press the CLEAR key once, and the number 5 will disappear from the edit line. Now press the CLEAR key again. The letter B on the edit line disappears, and so does the prompt Go to: Coordinate. You have backed out of the > or **GO TO** command completely, and you can now type something else.

Sometimes several characters will disappear from the edit line when CLEAR is pressed once. Type **123+** Now press CLEAR once. The + disappears. Now press CLEAR again. The whole number 123 disappears. Press CLEAR a third time and Value disappears.

There's an even faster but more drastic way to back out of a command. Press these keys: **>C12** Suppose you decide you don't want to use the **GO TO** command. Find the key labelled BREAK and press it once, watching the screen as you do so. The column borders flash and the prompt and edit lines are blank. You have backed out of the **GO TO** command with one key: BREAK. No matter what you are typing, you can always back out and leave the sheet unchanged by pressing the CLEAR key a few times or pressing the BREAK once, as long as you notice your error before pressing the last keystroke of the command or Ⓔ. Look at the screen. The VisiCalc program will tell you if you have made an error through messages or by flashing the column borders.

Before going on, spend a few more minutes moving the cursor around with the arrow keys, and the > or **GO TO** command. Try moving the cursor to a nonexistent position such as AB525. Giving an invalid coordinate to VisiCalc in the **GO TO** command causes the control panel of the screen to flash, and the cursor does not move.

Now, get back to the upper left hand corner of the sheet with **>A1**Ⓔ Next, we're going to learn how to write with the VisiCalc program.

Writing on the Electronic Sheet

As you have seen, moving the cursor and window around is pretty easy, but so far your electronic sheet is (or should be) empty. You'll find that writing on the sheet is even easier. Before proceeding further, type the following keys: **/CY** The screen will go blank, then reappear with the copyright on the prompt line. This will make sure that the sheet is clear and that the cursor is at position A1.

Now type the following keys using the SHIFT key to capitalize all letters: **SALES** (If you mistype a letter, you can back up with the CLEAR key.) Stop and look at the screen. On the prompt line is the word Label. This is the VisiCalc term for any type of alphanumeric message that won't be used in calculation which you write on the sheet. On the edit line is the word SALES, followed by the edit cursor. The edit cursor indicates that you can still use the CLEAR key to back up and retype the message, or to back out completely. SALES also appears at position A1 on the sheet. Your computer has the lower case alphabet, so all letters you type for labels will appear in lower case unless you use the SHIFT key for capitals, as you would on a typewriter. Use the CLEAR or the BREAK to back up and retype any characters in the SALES label, if necessary. When you are satisfied, press the ▶ key. The information on the prompt and edit lines disappears, and the cursor moves to position B1, leaving the label SALES at A1. (Throughout the rest of this tutorial, all alphabetic characters for labels will be shown upper case in the text. You can use lower case as you prefer for your labels—you won't hurt anything.) Try pressing the CLEAR key and then the BREAK key. (Nothing happens except the screen flashes.)

Now type the following keys: **100** Stop and look at the screen. The prompt line says Value, which is the VisiCalc term for a number or formula. On the edit line is the number 100, followed by the edit cursor. Press the CLEAR key four times, and watch the number disappear: First 0, then 0, then 1, then finally the prompt Value. Position B1 is still blank. You could, of course, have done the same thing by pressing BREAK.

Now press the following keys: 75+25. If you make a typing error, watch the edit line carefully and use the CLEAR key to back up. The word Value is again on the prompt line, and the edit line reads 75 + 25, followed by the edit cursor. Now press the ! key once. In place of 75 + 25, you now have 100 (the answer) on the edit line. You can use this "exclamation point key" feature to perform quick calculations before writing a number on the sheet. (Much more will be said about formulas and calculations later.)

Notice that, so far, nothing has appeared under the cursor highlight at B1. Everything has been happening on the edit line. Now press Ⓔ. The information on the prompt and edit lines disappears, and the number 100 appears at position B1 on the sheet. Try pressing CLEAR. (Nothing happens.) There is one other change on the screen since you pressed Ⓔ. The top line of the screen now reads: B1 (V) 100. This line is called the **entry contents line**, because it gives the full explanation of the contents written in the entry position highlighted by the cursor. Right now, the entry contents line says (V), for Value, and shows the number 100. Press the ◀ key, and the highlight cursor will move back to position A1. Now the entry contents line reads A1 (L) SALES. The (L) stands, of course, for label.

Formulas and Recalculation

Now move the cursor to position A2 by pressing the ↓ once. Next, type the letters **COST** and then press the → key. The cursor moves to B2, leaving the label COST at A2. Now we are going to write a different kind of formula. We want the entry at position B2, for COST, to be 60% of the number for SALES. Press the following keys: **.6*B1** The "*" symbol is used to indicate multiplication. The edit line should now read .6*B1 (If it doesn't, remember CLEAR.) Assuming that you are satisfied, press Ⓔ and watch what happens. The information on the prompt and edit lines disappears. On the entry contents line, you should now see B2 (V) .6*B1 And at position B2 is the number 60, the result of multiplying .6 times 100, the number at B1.

Press the ↑ key to move the cursor up to position B1. Now press the following keys: **200**Ⓔ and watch the screen. The number in the cursor highlight at B1 changes to 200. The number opposite COST, at B2, changes to 120. Notice that COST is still 60% of SALES.

Press the ↓ key, moving the cursor to B2. On the entry contents line at the top of the screen, the formula you had typed earlier, .6*B1 is still there. When you changed the number at B1 to 200, VisiCalc automatically recalculated the formula at B2 as .6*200, or 120. You'll see many more examples of this powerful recalculation feature as we progress.

More on Labels and Values

Let's look more carefully at labels and values, and explore an even simpler way to write formulas.

Press the following keys: **>A3**Ⓔ to move the cursor to position A3. We're about to write a formula to calculate gross profit as sales minus cost.

First, we'll write the label GROSS. Press the **G** key. Notice that the prompt line immediately says: Label. When you write at an entry position, the VisiCalc program looks at the first key you press to determine whether you are typing an alphabetic LABEL or a numeric VALUE. If you start with one of the letters **A** through **Z**, as you did here, the program assumes that you are typing a label. If you start with one of the digits **0** through **9** or a decimal point, ".", or with something that could begin a formula such as plus, "+", minus, "–", parenthesis, "(", "@" or "#" (which will all be explained later), the VisiCalc program assumes that you are typing a VALUE. For now, press BREAK to back out of LABEL.

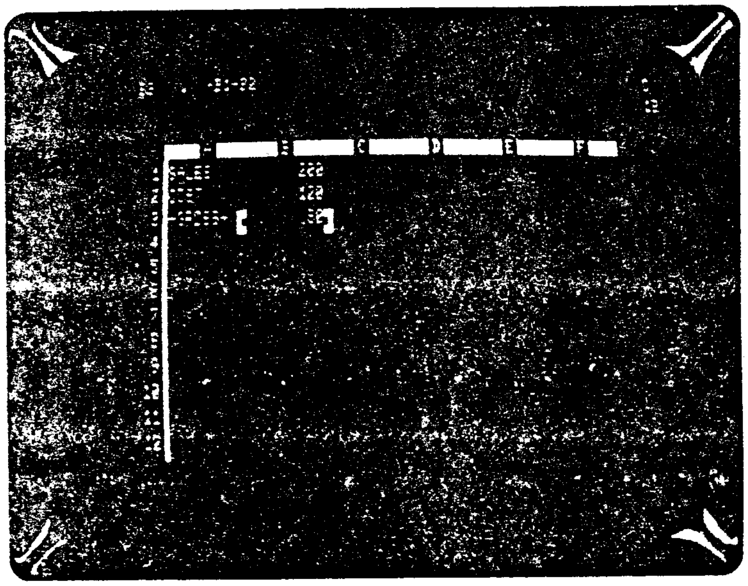
What happens if you want to write a message such as –GROSS– or 1ST QTR? Try it. Press the following keys: **–GROSS–** You may as well stop, because the VisiCalc program will be flashing the control panel at you. The initial dash (or minus sign) was taken to mean that you were typing a formula, the letter G as part of an entry position coordinate and then complained as you typed something that couldn't be a formula. Press BREAK to back out of this mistake.

To remedy this situation, press the " key (you have to hold down SHIFT). The VisiCalc program takes the " to mean that you want to type a LABEL, but this quote symbol will not be a part of the message itself. As soon as you press the " key, the prompt line says Label. The edit line shows just the edit cursor. Now you can type: –GROSS– As usual, you can use the CLEAR key to back up and correct mistakes. You don't have to type a closing quote symbol. Now press the ↓ key, and the cursor will move to position B3, leaving the message –GROSS– at A3.

Now we're ready to calculate SALES minus COST. The formula you'd expect would be B1–B2. What will happen when you type B1–B2? Try it: Type **B1–B2** No screen flashes so far, so press Ⓔ. What happened? If you've been watching the prompt line, you already realize that the VisiCalc program took the initial letter B to mean that you were entering a LABEL or message. B1–B2 is a perfectly legitimate message or comment, but it doesn't calculate anything. Under the cursor highlight you have, not the result of calculating SALES minus COST, but the message B1–B2.

Try again. First, we'll eliminate the erroneous message B1–B2. Try pressing CLEAR and then BREAK. Nothing happens, since we've already pressed Ⓔ. But we can use a new VisiCalc command to blank out an entry position at any time. Press the following keys: /BⒺ The message B1–B2 under the cursor highlight disappears, and the entry contents line goes blank except for the cursor coordinate B3. Note, however, that in most cases where you want to enter new information in a previously used coordinate, you can simply type in the new LABEL or VALUE without first blanking out the coordinate.

Remember that we used the " key to type a label when the VisiCalc program would otherwise have tried to treat it as a value. We can use a similar trick to type the formula B1–B2 as a value, instead of a label. One way to do this would be to type the formula as 0+B1–B2. The VisiCalc program would take the initial digit 0 to signify that we were typing a value. An equivalent formula is +B1–B2; the VisiCalc program will take the + to indicate a value. Press the following keys: **+B1–B2**Ⓔ Now we have what we want. The entry contents line reads B3 (V) +B1–B2. In the cursor highlight is the number 80, the result of calculating +B1–B2, or +200–120.



Cursor Moves in Formulas

So far, you know how to move the cursor to an entry position, and write a message, a number, or a formula which refers to other positions on the electronic sheet, such as +B1 – B2 in the previous example. By now you probably also realize that if you change the numbers at B1 or B2, the formula +B1 – B2 at entry position B3 will be recalculated, and the new result will be displayed at B3.

As you wrote the formula for sales minus cost, you probably had to check the screen to see that the number for sales was at position B1, and the number for cost was at B2. Now, imagine that you have already written a large number of formulas on the sheet. Keeping track of which numbers are at which coordinates would become time-consuming. Columns A and B might have scrolled off the left edge of the window as you moved it to display other parts of the sheet. Then you wouldn't even be able to see the numbers next to SALES and COST, so writing a new formula involving sales and cost might be a problem.

There's an easy way to solve this problem: you write the formula, but let the VisiCalc program fill in the coordinates! First, blank out the formula at B3 by typing: /BⒺ.

Press the + key. The prompt lines reads Value, and the + appears on the edit line, followed by the edit cursor. At this point, we would normally type the coordinate B1, but what we really want is the number next to the label SALES (currently 200). Try pointing at that number on the screen with your finger. We are about to do the same thing electronically, by pointing with the cursor.

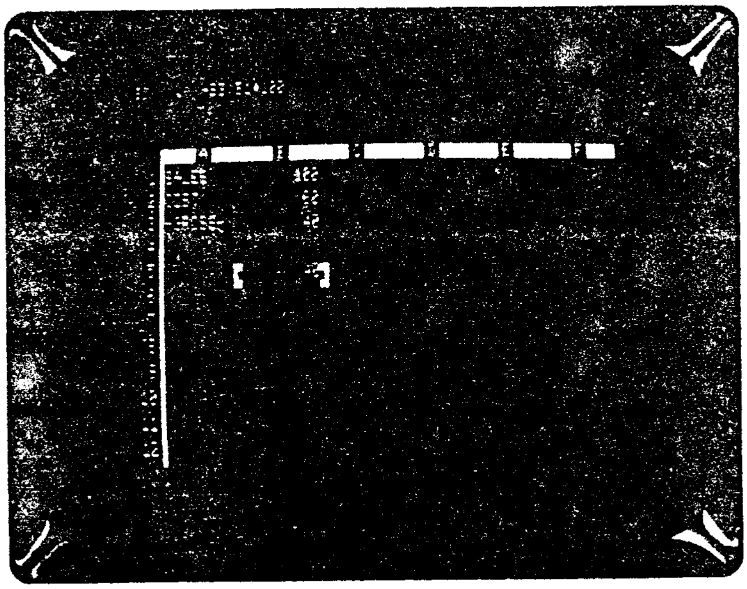
Watch the edit line and press the ↑ key once. The cursor moves up to highlight the number 120, and the coordinate of that number, B2, appears on the edit line. Now press ↑ again. Do you see what we mean by "pointing" the cursor? The edit line now reads +B1 followed by the edit cursor. Now press the – key. The cursor jumps back to B3, the entry at which we are writing the formula. The edit line now reads +B1 – followed by the ubiquitous edit cursor. In general, after pointing the cursor at the position you want to include in the formula, you simply continue the formula by typing an arithmetic operation symbol such as – (minus), + (plus), * (multiplication), or / (division).

Watch the edit line again, and press the ↑ key once more. The cursor moves up to 120, the number next to COST, and the edit line now reads +B1 – B2. This is the formula we want! To end the formula as it stands on the edit line, press Ⓔ The information on the prompt and edit lines disappears; the entry contents line reads B3 (V) +B1 – B2; and under the cursor highlight is the number 80, the result of calculating B1 minus B2.

To demonstrate the VisiCalc program's recalculation feature again, press the ↑ key twice to move the cursor up to B1, and change the number there by typing: 100Ⓔ Notice that B2 changes back to 60, or .6 times 100, and B3 changes to 40, or 100 – 60.

You can always type the position coordinates yourself instead of pointing the cursor. But as you gain experience and familiarity with the VisiCalc program, you'll probably find that the technique of moving the cursor to the positions you want will become easier and more natural. In time, you'll find that you can almost forget about coordinates entirely, and think in terms of the visual positions of numbers and formulas on the sheet.

To test your understanding of the process of moving the cursor as you write formulas, you may wish to try an example yourself. Move the cursor down to position B5, and write a formula there for gross profit as a percentage of sales. (Hint: The desired formula is +B3/B1•100. Try to obtain this formula by moving the cursor and typing the + and / keys.)



More on Editing

Suppose we want to change the cost in the example we are working on from 60% to 55%. Place the cursor on B2. The entry line shows B2 (V) .6*B1. To change that formula type **/E** The prompt line reads [Edit]: Value (because the entry is a value). The formula .6*B1 is displayed on the edit line with the edit cursor on the first character, the decimal point. We want to change .6 to .55. Press the ▶ twice to move the edit cursor on the *, just to the right of the part we want to change. Nothing changes except the position of the edit cursor. Now press the **CLEAR** key once. This works must like it did when we erased with the **CLEAR** key before. The edit cursor backs up one space and erases the 6. The edit line now reads .*B1. Type **55** The edit line reads .55*B1—the formula we want. Press Ⓔ The prompt and edit lines clear and B2 (V) .55*B1 appears on the entry line. The number at B2 has changed to 55. To see if you have learned the technique, change the 55 back to 6.

Pressing **CLEAR** will erase the character just to the left of the edit cursor. Typing a character will put that character on the edit line just in front of (to the left of) the edit cursor.

The right and left arrow keys will move the edit cursor one position to the right or left without changing the characters on the edit line. The up arrow key will move the edit cursor to the beginning of the entry. The down arrow will move it to the end of the entry.

The Edit command operates in two ways. Typing **/E** will let you edit the contents of any position on the sheet. Typing **SHIFT–ENTER** (by holding down **SHIFT** while pressing Ⓔ will let you edit what is on the edit line before you enter it into an entry position.

You can use this latter editing feature on anything you are typing in on the edit line. Type **>A6** Ⓔ **JANUARY PROFIT MARGIN** Suppose you need the profit margin for February instead of January. Press **SHIFT–ENTER** This time the prompt line reads [Edit]: Label because we are entering a label. Press the ◀ key until the edit cursor is resting just to the right of the Y. Press **CLEAR** seven times to erase January and type **FEBRUARY** If you were to press Ⓔ you would have February Profit Margin entered at A6. For now, return to the original sheet by pressing **BREAK** and typing **>B1** Ⓔ For more information on the Edit command see Part III, the EDIT Command.

Saving the Electronic Sheet on Diskette

As we wrap up this first lesson, you'll have a chance to use one of the storage diskettes you created when you read the section entitled "Initializing VisiCalc Storage Diskettes" in Part I. We'll save the contents of the electronic sheet from this lesson on diskette, and load the sheet back into memory as we begin Lesson Two. If you skipped the discussion of diskette initialization when you read Part I, the Introduction, you can save your sheet on your working copy of the VisiCalc diskette for now, by following the instructions below. After your sheet is safely saved, go back to the section entitled "Initializing VisiCalc Storage Diskettes" and make some. If you don't have any blank diskettes available right now, get some from your Computer Center as soon as possible. You may also initialize previously used diskettes which contain data you no longer want.

For insurance, in Lesson Two we'll give brief instructions for setting up the electronic sheet with exactly the same labels, numbers and formulas which you have now.

The VisiCalc program can also write to other devices such as printers or modems. See **the PRINT command in the Reference Section of this manual.**

If you have an initialized storage diskette, insert it into drive 0. Gently push the diskette all the way in, and close the drive door pushing the handle down. If you haven't made any VisiCalc storage diskettes, save your sheet on your VisiCalc working copy diskette in drive 0 for the time being. You must have the write protect notch uncovered before you can save to the diskette.

Now type the VisiCalc STORAGE command **/S** The prompt line reads `Storage: L S D Q #` The VisiCalc program is telling you that it recognizes **/S** as the storage command, and that next it expects you to press one of the keys **L S D Q** or **#**. These keys have the following meanings:

L	Load the sheet contents into the computer's memory from a diskette file.
S	Save the sheet that is in memory by "writing" it into a diskette file.
D	Delete a previously saved file from its place on the diskette.
Q	Quit the VisiCalc program.
#.	Save or load a sheet in the DIF (Data Interchange Format) (see Part III, the STORAGE Command).

Press the letter S. Now the prompt line reads Storage: File for Saving, and the edit cursor has appeared on the edit line. There is room on the diskette to hold several electronic sheets. So that you can find the particular sheet you want later, you give each sheet a name when you save it on diskette. The saved information is called a diskette file, and the name you give it is called a **file name.** The VisiCalc program will find an empty area on the diskette and will write the sheet contents there. Then, in an area on the diskette called its directory, the VisiCalc program will save the file name you specified and a note about where on the diskette it saved the sheet's contents.

When you are saving a file, the first thing you must tell the VisiCalc program is **the file name.** A file name can be up to eight alphanumeric characters beginning with a letter. After the file name, you will add the characters /VC. This allows you to differentiate the VisiCalc files saved with the command **/SS** from any others you may save on the diskette. See a complete discussion of file names in your *TRS-80 Model III Owner's Manual.*

Let's name the file we are saving EXAMPLE/VC. Since we instructed you to place the storage diskette in drive 0, the VisiCalc program will save the file on the diskette in that drive. **Press the keys: EXAMPLE/VC** As usual, you can correct typing errors with the CLEAR key. When you are satisfied, press Ⓔ. The disk drive will begin running and the ACTIVE light will come on. After a moment, the drive will quiet down, the ACTIVE light will go off, and the prompt and edit lines will go blank. Your work is safely filed away on the storage diskette.

This completes Lesson One. You may wish to experiment for a while, moving the cursor around and writing your own labels, numbers and formulas. Try writing some formulas by pointing the cursor to obtain the coordinates. There are, of course, many more features of the VisiCalc program which we haven't discussed yet, and you may stumble upon one of them. As you experiment, if something happens that you don't understand, make a note of it, and then continue with the next lesson. Most of your questions will be answered as you go. Already, though, you know enough about the VisiCalc program to use it for some simple applications. Try it!

Postscript: Protecting Your Work

Have you ever worked out a problem or made some notes to yourself on a sheet of paper, only to find out later that the sheet had been lost, or that someone had accidentally thrown it away? Or perhaps you've lost the results of some calculation you were performing on an electronic calculator, because power was turned off, or the battery died. Things can and do go wrong.

As you begin to use the VisiCalc program, you'll find that at times the results you see on the screen may be quite important to you. Losing that information at the wrong time would be at best a real nuisance, and at worst you could lose data that could take hours to recreate, if, say, you needed the results for a presentation the next morning. How can you protect yourself?

The VisiCalc program is designed to be as foolproof as possible. **It is quite difficult to mess things up badly by anything you might type at the keyboard.** For example, if you give the command to clear the screen (intentionally or accidentally) by typing /C, the VisiCalc program will display on the prompt line CLEAR: Type Y to confirm. Only if you type the letter Y at this point will the contents of the sheet be erased. If you type any other key, the clear command will be "aborted", and the sheet will be left unchanged. Similarly, if you type /B to blank out an entry, nothing will happen unless and until you press one of the keys ▶, ◀, ▼, ▲, or Ⓔ. Any other key will abort the blank command.

Save Your Work as You Go

The VisiCalc program does its best to protect you, but other things can go wrong. What if your building has a power failure? What if someone pulls the plug from the socket? What if you are called away by some emergency and the janitor turns off your computer?

To protect yourself, you should save the electronic sheet periodically on diskette. As you work, think of how long it has been since you last saved the sheet. If you have spent more time than you would wish to lose if something went wrong, or if you have new results which might be difficult to reconstruct, then it's time to save the sheet again. To keep track of several versions of the same information on diskette, you can append a sequential number (such as FORCAST1/VC) to the file name you use when you save the sheet. Remember that only eight alphanumeric characters are allowed for the file name and spaces may not be used.

Another option is to print out the sheet on a printer. (See the discussion of the print command at the end of Part II, Lesson Three and in Part III, The PRINT Command, of this manual.) Although you would not be able to examine the formulas, or change the numbers and recalculate, this "hard copy" would at least survive through most interruptions and emergencies. You can, however, print the formulas with the /SS:P command. See the STORAGE command.

Making Backup Copies of Diskettes

Saving your work periodically on diskette is only the first step in protecting yourself. A diskette is a safe and reliable medium for storing information; however, to be used, a diskette must be handled and transported from place to place. Think pessimistically. A diskette may be scratched, or it may pick up grease or dust; it may be damaged by heat, exposed to a magnetic field, or accidentally reinitialized (which erases its contents). And a diskette will eventually wear out. Hence, to protect yourself, you should always make extra copies of your important files on separate "backup" diskettes.

To make a backup copy of your files, you must leave the VisiCalc program. Save the sheet you have been working on, then type /SQ. The prompt line will read: Quit: Type Y to confirm. Press Y. After a few seconds your screen will display: TRSDOS READY. Follow the same procedure you used to make copies of your VisiCalc storage diskettes, described in Part I, the section entitled "Instructions for Using BACKUP". The TRSDOS BACKUP command will copy the entire contents of your storage diskette onto a blank diskette, so you have an identical backup copy of the original. The TRSDOS BACKUP command is also discussed in the *TRS-80 Model III Owner's Manual* that comes with your computer. *Test your understanding of the procedures on a data diskette you make expressly for experimentation.* Until you thoroughly understand the procedures, don't risk using diskettes containing valuable data; you may lose the data while learning.

Making backup files is important. It's all too easy to read about these protective measures, use them once or twice, and then when you're in a hurry, skip the backup steps. When disaster strikes, you'll curse your carelessness. The time it takes to make backup files will be amply repaid the first time you try to load a file from a diskette, only to get the message that an error occurred while trying to read the disk.

Remember Murphy's Law:

> If anything can possibly go wrong, it will.

Protect yourself.

Lesson Two

Lesson Two

In this lesson, the "fireworks" begin as we show how the VisiCalc program extends and generalizes the basic principles you have seen so far. We will begin with the example built up in Lesson One, showing a figure for SALES, and formulas to calculate COST of goods sold and GROSS profit.

If you have just finished Lesson One, everything you need should still be present on the electronic sheet. Check your screen against the screen photo below, and continue with the text from that point. If you want to practice loading the sheet you just saved, anyway, type **/CY** and then follow the instructions in the next section.

If you're tackling Lesson Two in a new session, your first step is to load the VisiCalc program into your computer's memory. The instructions to do this are in Part I in the section entitled "Loading the VisiCalc Program."

If you saved the results of Lesson One on diskette, you can easily reload the same information now by following the instructions below for "Loading the Sheet from Diskette". If you don't have the file EXAMPLE/VC saved on diskette, you can type in the same information from the keyboard. To practice moving the cursor and writing labels, numbers and formulas, you can go through the steps in Lesson One, or, if you're comfortable with these VisiCalc features, you can type exactly the characters you see below. (All photo examples in this manual will show labels in upper case. You can type your labels in lower case if you choose and your screen will show them in lower case.)

```
>A1Ⓔ
SALES♦100Ⓔ
>A2Ⓔ
COST♦.6*B1Ⓔ
>A3Ⓔ
"-GROSS-♦+B1-B2Ⓔ
>B1Ⓔ
```

Now go on to "Replicating a Formula".

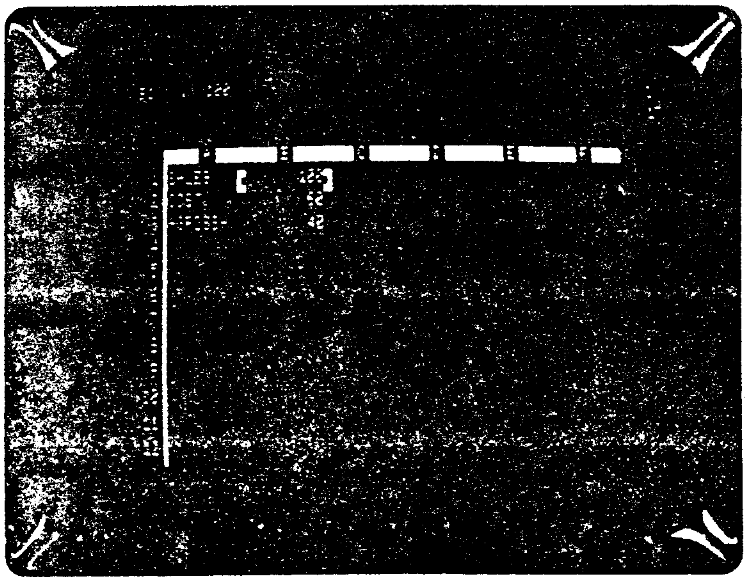
Loading the Sheet from Diskette

Make sure that your disk drive's ACTIVE light is off and/or the drive has stopped running, then open the drive door. Now insert the storage diskette holding the file name EXAMPLE/VC which you saved at the end of Lesson One into drive 0.

Now type the VisiCalc STORAGE command /S. The prompt line reads STORAGE: L S D Q #. (To review the meanings of the characters L S D Q #, check the section entitled "Saving the Electronic Sheet on Diskette" at the end of Lesson One.) Press L. The prompt line reads Storage: File to Load. At this point, you could simply type the name of the example file, but try something else. Press the ◆ key once. Your disk drive will run for a moment as the VisiCalc program looks at the directory and the ACTIVE indicator light on your drive will come on. To see a full description of the use of the ◆ key to find a file on your diskette, see the Storage command in the Reference section of this manual.

Then the name EXAMPLE/VC:0 appears on the edit line! (The :0 is the number of the disk drive containing the storage diskette.) If a different name appears, press the ◆ key repeatedly until you get the name EXAMPLE/VC. When you press the ◆ key, the VisiCalc program looks in the diskette's directory or catalog for the name of a file which could be a saved VisiCalc sheet. On the edit line, the VisiCalc program presents the first qualifying file name it finds in the directory for your inspection. If this file name is not the one you want, you would simply press the ◆ key again and the VisiCalc program would show you another file name from the diskette directory. Eventually you will reach the name of the desired file or else you'll realize that you have the wrong diskette.

Assuming that you have the name EXAMPLE/VC on the edit line, press Ⓔ Watch the characters flashing by at the left end of the edit line as the file is loading. These are the same keystrokes you would type from the keyboard to set up the sheet, but they are being "typed" automatically at high speed as they come back from the diskette. After a moment, the disk drive ACTIVE light goes off, the information on the prompt and edit lines will disappear, and the screen should look just like the screen photo below. You can continue with the instructions following the photo.



Replicating a Formula

Your screen should resemble the one pictured above. The cursor should be at B1. If the number under the cursor is not 100, just type **100** and press Ⓔ

At present, we have figures for sales, cost of goods, and gross profit for only one month (or year, or other period). Now, we'll project these figures out for twelve months. Let's assume that sales will increase by 10% each month. Press the ◆ key to move the cursor to C1, and type: **1.1*◆**Ⓔ Notice that we used a cursor movement to fill in the coordinate B1 in the formula for the next month's sales. The entry contents line at the top of the screen now reads C1 (V) 1.1*B1, and the number in the cursor highlight at C1 is 110, or 1.1*100

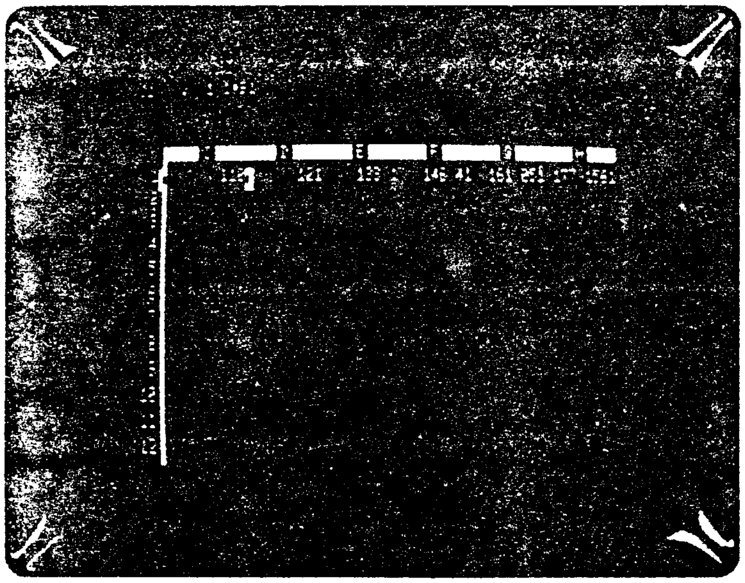
To calculate sales for the remaining ten months, we would have to move the cursor to D1 and type the formula 1.1*C1, then move on to E1 and type 1.1*D1, and so forth. Since this is such a common operation and it requires so many keystrokes, the VisiCalc program provides a shortcut way to do it. The shortcut is the **REPLICATE** command, and it can be used to make copies of, or "replicate" formulas, labels, numbers, blank entries, etc., across columns, down rows and so forth. In this lesson, we'll use the REPLICATE command in just a couple of simple examples. The many uses of this command are covered more fully in Lesson Three.

With the cursor at C1, type **/R** The prompt line reads Replicate: Source range or ENTER, and on the edit line is C1, the coordinate of the formula under the cursor, followed by the dash. Press Ⓔ The prompt line now reads Replicate: Target range, and the edit line reads C1 ... C1: followed by the edit cursor. So far, we have indicated that we want to replicate just the formula at C1, and the VisiCalc program is asking us where we would like to put copies of this formula.

Our intent is to project sales out for twelve months. The first month is shown at B1, the second at C1, and the twelfth month will be at M1. Hence, we want the formula replicated in the range of positions from D1 to M1. Press the ◆ key, then type a period. The edit line now reads C1 . . . C1: D1 . . . As you can see, by moving the cursor, we are indicating where we want copies of the formula to be placed. Now press the ◆ key ten times. Notice that as you move the cursor to D1, E1, F1, and so on, the "target range" on the edit line reads D1 . . . D1, D1 . . . E1, D1 . . . F1, and so on: the VisiCalc program is "filling in" the ending coordinate of the target range, just as it did when we moved the cursor in the middle of typing a formula.

When you have finished pressing the ▶ key ten times, the cursor will be resting at M1, and the edit line will read C1 . . . C1: D1 . . . M1 (If you moved too far with ▶, you can move back with ◀.) Now press Ⓔ The cursor jumps back to C1, where the original formula is stored. The edit line reads C1: D1 . . . M1: 1.1*B1 and the edit cursor follows the coordinate B1 on the edit line. The prompt line reads Replicate: N=No Change, R=Relative The VisiCalc program is asking whether we want the same formula, 1.1*B1, at each of the positions in the target range D1 . . . M1, or whether the coordinate B1 should be interpreted as *relative* to the position of the formula.

We want sales to increase by 10% in each month, so we want the formulas to be 1.1*B1, 1.1*C1, 1.1*D1, etc. In other words, each new sales figure should be 1.1 times the previous month's sales. Press **R** and watch the screen. The information on the prompt and edit lines disappears; the cursor remains at C1, where we started; and numbers have appeared in the visible columns.



Use the → key to move the cursor to D1, E1 and F1, and notice what appears on the entry contents line: 1.1*C1, 1.1*D1, and 1.1*E1. Press the → key seven more times to move the cursor over the entries G1 through M1, scrolling the window to bring them into view. The VisiCalc program has "typed" the formulas for you, and calculated the sales values for all twelve months. At M1, the twelfth month's sales should be shown as 285.3117 (to four decimal places).

Much has happened during the last few keystrokes, and of course there are many more options for the REPLICATE command which we haven't used in this case. But what you must remember to type to replicate a formula is very simple:

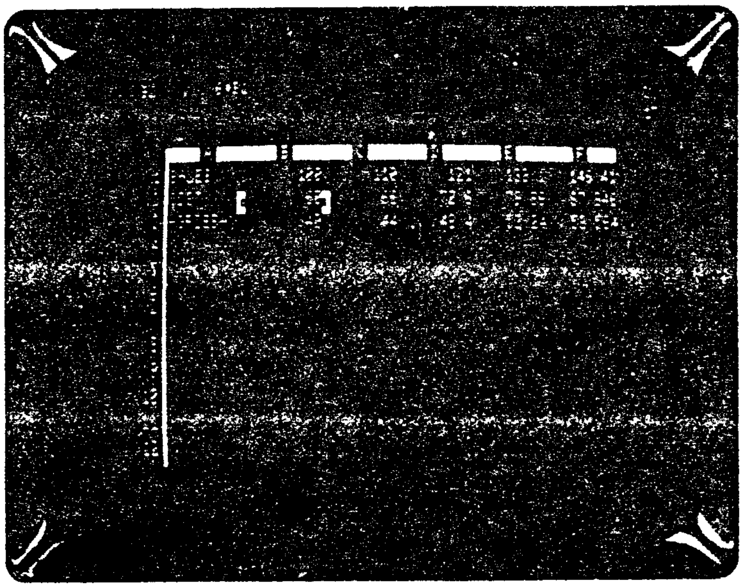
1. **Point the cursor at the first entry position in the range you want to replicate;**
2. **Press /R to start the** REPLICATE **command;**
3. **Press . and indicate the last entry position in the range to be replicated and then press Ⓔ or :**
4. **Point the cursor at the first entry position in the range of positions where you want the formula replicated, press . and point to the last entry position (or you can type in the entry coordinates); and**
5. **For each coordinate in the formula, press either N or R depending on whether that coordinate should be left unchanged, or interpreted as relative to the position of each copy of the formula.**

Replicating a Range of Formulas

Type >**A2**Ⓔ to scroll the window all the way back to the left edge of the sheet, and then press ▶ to move the cursor to B2. To complete our twelve month projection, we would like to replicate the formulas for cost of goods sold and gross profit. At the moment, the entry contents line shows the formula for cost of goods, .6*B1 If we were to move the cursor to B3, we would see the formula for gross profit, +B1−B2 We can replicate both of these formulas at once across the sheet.

Press /R The prompt line reads Replicate: Source range or ENTER and on the edit line is B2, followed by the edit cursor. If you were to press Ⓔ at this point, as we did before, you would replicate just the formula for cost of goods sold at B2. When you were finished, you would come back to the formula for gross profit at B3, and replicate it into the same range of columns as you did for B2. We can accomplish both of these steps at once: **Press .B3**Ⓔ The edit line now reads B2 . . . B3 The VisiCalc program acknowledges that we want to replicate a "source range" of formulas B2 through B3. The prompt line reads Replicate: Target range Instead of pointing at the first and last positions as we did before, we will simply type the coordinates of the target range. Type **C2.M2**Ⓔ For a source range of B2 . . . B3, and a target range of C2 . . . M2, VisiCalc will assume that the formula at B2 (for cost of goods) is to be replicated at positions C2 through M2, and the formula at B3 (for gross profit) is to be replicated at positions C3 through M3.

The prompt line now reads Replicate: N=No Change, R=Relative and the edit line reads B2: C2 . . . M2: .6*B1 with the edit cursor following coordinate B1. How do we want the coordinate B1 to be interpreted? Our intent is that cost of goods sold in any given month will be 60% of that month's sales. Hence we want B1, the coordinate for SALES, to be interpreted as *relative* to the position of each copy of the formula. For the formula at B2, we want the sales figure just above it, or B1. For the formula at C2, we want the sales figure just above *it*, or C1; and so on. Press **R** Almost instantly, numbers appear in the other columns, and the edit line now reads B3: C3 . . . M3: +B1 with edit cursor following B1. The VisiCalc program is ready to replicate the formula for gross profit, at B3, into positions C3 through M3. The +B1 is the beginning portion of the formula +B1−B2. Again, we want B1 to be interpreted as relative to the position of each copy of the formula. Press **R** Now the rest of the formula appears on the edit line, as +B1−B2 this time with the edit cursor following B2. B2, or cost of goods, is also relative in the gross profit formula. Press **R** once more. Numbers will appear in the other columns of row 3, and the prompt and edit lines will go blank. The REPLICATE command has finished its work.



Use the arrow keys to move the cursor to the right and up and down to examine the formulas and calculated results displayed in columns C, D, E and so on. Finally, type **>M1Ⓔ** to display the last month's sales, cost of goods, and gross profit in column M. The VisiCalc program has saved you a good deal of work already. But these numbers are somewhat hard to read, because they fill the columns and don't always line up. Can we do better than this?

Formatting the Screen Display

Of course. Type **/GFI** (for "global format integer," as explained below), and watch what happens. Is the new display easier to read? If you scroll the window to the left, you will see that all of the numbers which come into view on the screen have been rounded to integers and lined up on the right. However, this does not mean that the VisiCalc program has actually rounded the numbers that it uses in its calculations. Each number is calculated and maintained with up to eleven significant digits or decimal places (so that, for instance, each new period's sales is based on an accurate rendition of the previous period's sales). The numbers are rounded only as they are displayed on the screen.

Perhaps you'd prefer to see two more decimal places, for "dollars and cents." Press / the keystroke which starts all commands. The prompt line reads Command: BCDFGIMPRSTVW—. Each of the keys, B, C, D, and so on through '—' is the keystroke for a different command. So far, we have seen the commands **/B** (for **BLANK**), /C (for **CLEAR**), **/R** (for **REPLICATE**), and /S (for **STORAGE**). Now, press G. The prompt line reads Global: C O R F. The VisiCalc program has recognized the GLOBAL command, which is used to change something about the entire screen display. Next, the VisiCalc program expects one of the keystrokes C, O, R or F. Press F. Now the prompt line reads Format: D G I L R $ *. The VisiCalc program is ready for a change to the "global format," or the way in which numbers and alphabetic labels are displayed on the screen. A moment ago, you used the letter I to change all of the numbers to integer format. **Now, press $.** Remember that you can use the CLEAR key to stop the command and start over. Notice how the screen display changes to show everything in dollars and cents form.

As you might guess from the FORMAT prompt list D G I L R $ *, there are many other ways to format numbers and alphabetic labels. It is also possible to set the format of each entry position individually. These options will be covered more fully in Lessons Three and Four of this Tutorial.

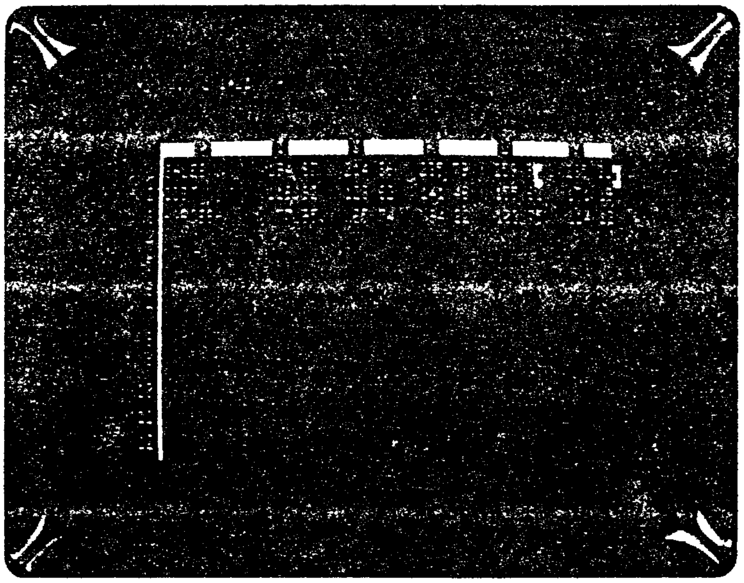
Fixing Titles in Place

If the cursor is not on M1, type **>M1Ⓔ** Only the numbers for each month's sales, cost of goods and gross profit are shown on the screen. The titles SALES, COST and –GROSS– have scrolled off the screen window to the left. Imagine the situation if you were preparing a more complex income projection, with many rows of numbers for selling and administrative costs, taxes, and so on. It would be difficult to remember what each row of figures represented, once the titles had scrolled off the screen. We'd really like the titles to stay visible at the left edge of the screen window.

No more easily said than done. Type **>A1Ⓔ** to bring the titles SALES, COST and –GROSS– into view. Now type **/T** (the **TITLES** command). The prompt line says Titles: H V B N The possible keystrokes are:

H	To fix horizontal titles.
V	To fix vertical titles.
B	For both horizontal and vertical.
N	For neither.

For now, press **V** You have asked the VisiCalc program to fix the vertical column A, where the cursor lies, in its present position, no matter where the cursor is moved subsequently. Press the ▶ key eight times, watching the screen as you do so. Notice that column A stays fixed in place, while the remaining columns scroll to the left, disappearing when they reach column A. Now press the ◀ key nine times. (If you aren't sure of what happened, press the ◀ key again.) You are "bumping" into column A, just as you bumped into the left edge of the sheet once before. Next, type **>M1Ⓔ** Column A is still visible, making it easy to identify each row of numbers.



Now, for a Quick Recalculation

So far, with the aid of the REPLICATE command, you have written one number (the beginning number for sales at B1), and thirty-five formulas on the electronic sheet. How are these formulas related? Press **>B1Ⓔ** to bring the first columns into view, leaving the cursor on the initial sales figure. The formula for cost at B2 is .6*B1 which depends on the figure for sales at B1. The formula for gross profit at B3, in turn, depends on both sales and cost (+B1−B2). What about succeeding columns? At C1, the formula is 1.1*B1 so this entry also depends on the initial sales figure. And cost of goods at C2 depends on the figure at C1, while C3 depends on both C1 and C2. At D1 we have 1.1*C1, and so on. As you can see, a change to the initial sales figure at B1 will affect every other number on the sheet. Try it. Just type a new number, such as 123.45, and press Ⓔ How long did it take to recalculate twelve months' worth of sales, cost of goods and gross profit formulas? Type **100** (and Ⓔ) again, and watch the changes ripple through the other figures on the screen. Scroll the screen window to the right to view M1, the ending month's sales.

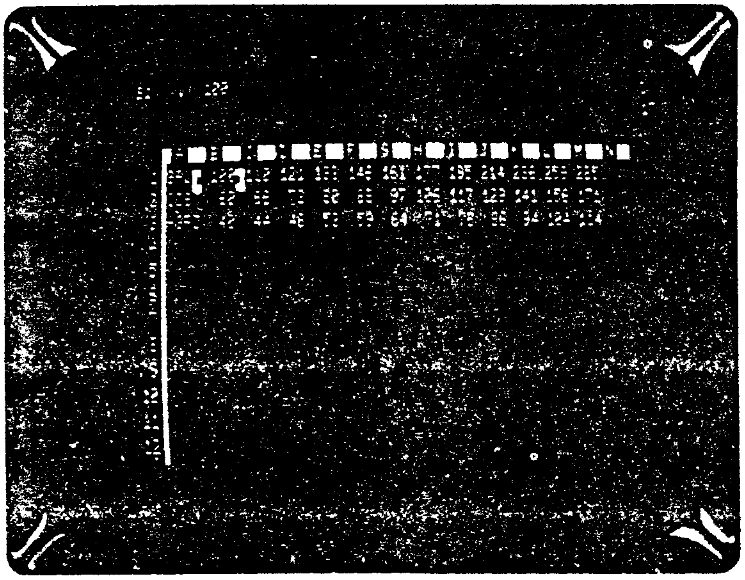
At this point, if you're intrigued by this example, you'd like to know how to change the sales growth figure of 10%, or the cost of goods percentage of 60%, and recalculate gross profit. For simplicity's sake, we've designed this example so that the only changeable figure is the initial sales. Since the figures 1.1 and .6 are built into each of the twelve formulas for sales and cost of goods, we can't change these percentages without replicating all of the formulas again. A better approach would be to write the factors .1 and .6 into separate positions on the sheet, and make the sales and cost of goods formulas refer to these positions. If we did this, changing the sales growth and cost of goods percentages would be as easy as changing the initial sales. We'll make use of techniques like this in Lesson Three. (If you're *not* intrigued by this business-related example, bear with us; Lesson Three deals with personal budgeting.)

Adjusting Column Widths

Type **>B1Ⓔ** For some time now, you've been scrolling the window back and forth across the sheet to view the figures for different months. You might be wishing for a larger screen that would display more columns at once. Well, you *can*.

Type another GLOBAL command: **/GC7Ⓔ** In an instant, the screen changes to display more columns. Each column has been narrowed from nine characters to seven characters in width. You can use the command **/GC** to set the "global column width" to anywhere from 3 to 60 characters per column. Given a column width, the VisiCalc program will fit as many columns as it can across the screen, placing a blank to the left of each value so that the numbers do not run together from one column to the next.

Right now, the dollars and cents figures just about fill up the available space in these 7-character columns. But we already know how to round the numbers to integers: Type **/GFI** Now, we have some extra space, so let's narrow the columns further: Type **/GC4Ⓔ** That gives us even more columns of figures in the window, each column four characters wide.



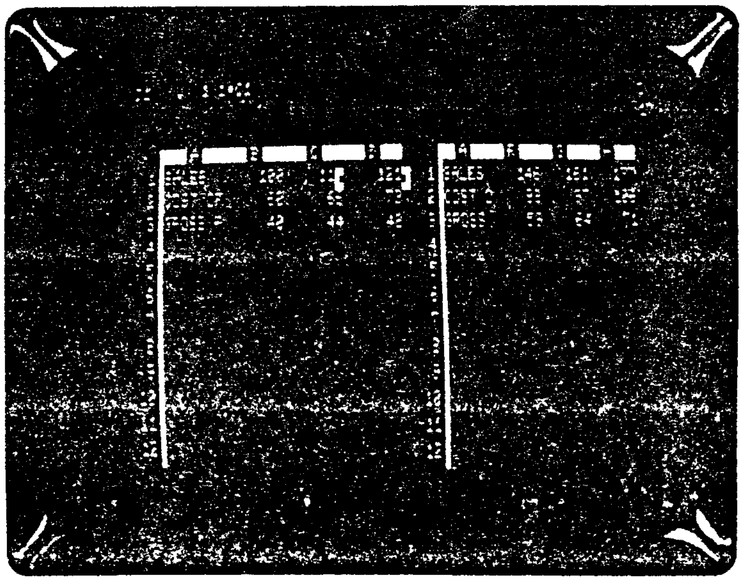
Pause for a moment. Look at the titles, SALES, COST, and –GROSS– in column A. SALES has been shortened to SALE, and –GROSS is now –GRO Have we lost the rest of the labels SALES and –GROSS–? Move the cursor to position A1 to find out. If you press the ← key, you'll bump into column A, which we fixed in place as a title area. Type >**A1**Ⓔ The entry contents line reads A1 (L) SALES Press ↓ twice to reach A3. The entry contents line reads A3 (L) –GROSS– Even though the columns have been narrowed, the full alphabetic labels are preserved.

Can these labels be more than nine characters long? Of course; in fact, you can type an alphabetic label as long as 125 characters, regardless of the current column width, and the full label will be preserved. Try this: with the cursor still at A3, type **GROSS PROFIT** Then press the ↑ key to move up to COST, and type **COST OF GOODS SOLD** followed by Ⓔ Next, type /**GC12**Ⓔ Notice that the entire label GROSS PROFIT comes into view, while the number of columns in the window is reduced. Now, type /**GC18**Ⓔ Notice the prompts Global: C O R F and Column width. As usual, the CLEAR key can be used for corrections as you enter the number 18. The entire label COST OF GOODS SOLD can be seen. Let's go back to /**GC7**Ⓔ

The ability to fix titles in place and adjust column widths gives us some compensation for the limited size of the screen. But, suppose we'd really like to keep the wider columns and the cursor at B1, the initial sales figure, and change it, while watching what happens to the final sales and gross profit at column M. If only we had two screens . . . .

Splitting the Screen

Type >**E1**Ⓔ to move the cursor to column E. Now, type the **WINDOW** command /**W** The prompt line reads WINDOW: H V 1 S U Press **V** (for vertical). Your screen should resemble the photo below.



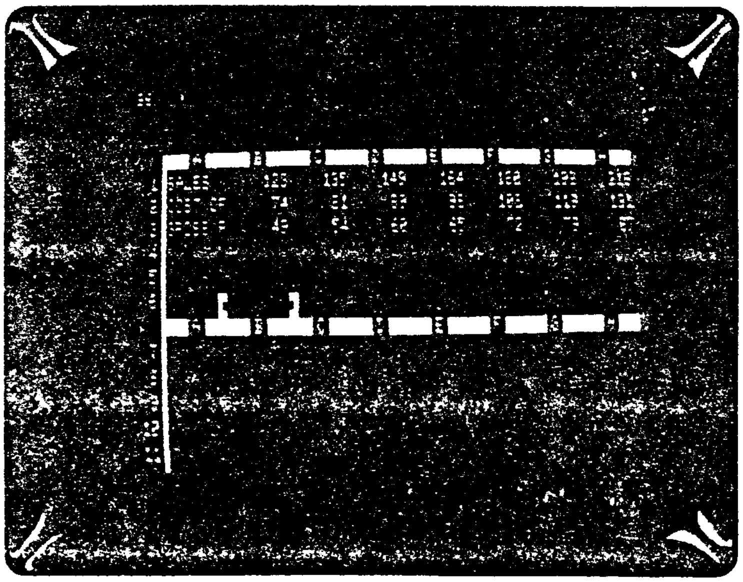
You have created two screen windows. Each one can be scrolled independently to view any portion of the electronic sheet. At present, the cursor is in the left hand window. Use the ◆ key to move the cursor downwards to row 12 and beyond. The left window will scroll downwards to follow the cursor, but the right window will remain still. Bring the left window back to the top of the sheet with >B1Ⓔ

Now press the ; key. The cursor jumps into the right hand window. Press the ◆ key to scroll the right window across to column M. We now have both the beginning and ending months' sales, cost of goods and gross profit figures visible at the same time. Press ; again. The cursor jumps back to the left window. (Each time you press the ; key, the cursor jumps from one window to the other.) Notice that the cursor has landed at the same position it was on when we last left the left hand window.

Now we can change the initial sales figure, and watch what happens in the final month. Type **123** followed by Ⓔ and let the VisiCalc program recalculate. Sales in column M should be 351 Try typing a few more numbers with the cursor at B1. See if you can find, by trial and error, the initial sales figure which gives you an ending month's sales of 1000 (rounded to the nearest integer). (Hint: You can type a number with a decimal point, even though it will be displayed in rounded form.)

When you are finished experimenting with recalculation, type the WINDOW command /W again. This time, press **1** The screen returns to normal (1 window). We have seen that the idea of a split screen is useful. Can we get any more mileage out of this approach? Take a look at the screen, and notice how much of it is empty. Perhaps we can use the lower part of the screen to better advantage.

Type >B7Ⓔ to move the cursor down to the middle row of the screen. Now type **/WH** (for horizontal). Your screen should look like the photo below.

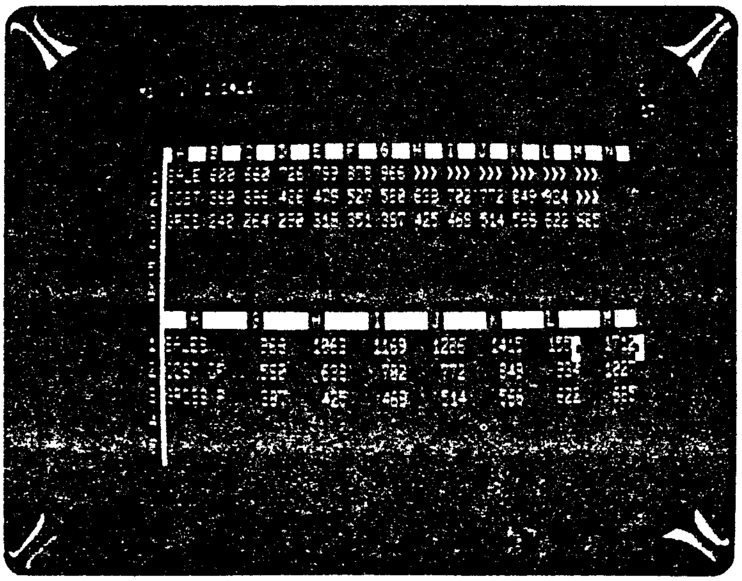


This time the screen is split horizontally, into a top and bottom window. Press the ; key to move the cursor into the bottom window. Next, hold down ▲ to scroll the window upwards, until the cursor bumps into the top edge of the sheet. Both windows are now displaying the same portion of the electronic sheet! The cursor should be at B1 in the bottom window. Change the number at B1 to 100 in the bottom window, and watch what happens. The recalculation affects both windows. Use the ▶ to scroll the bottom window rightwards until column M comes into view. Now we can see the first and the last months at the same time.

Global Commands in Separate Windows

Press ; to move the cursor into the top window, and type /GC4Ⓔ Once again, we have more columns on display at the top of the screen, each column four characters wide. As you can see, the column widths can be different in the two screen windows. The global commands /GC and /GF affect only the window in which the cursor rests at the time the command is typed. Press ; once more, moving the cursor to the bottom window, and this time type **/GF$** We have dollars and cents displayed in the bottom window, and integers at the top.

Press ; to jump the cursor into the top window, and type >**B1**Ⓔ to highlight our original sales figure. At B1, type the number **300** followed by Ⓔ and watch the changes ripple through all the columns as the VisiCalc program recalculates all the formulas. Now type the number **600**Ⓔ What happens? Columns H through M show > symbols in some positions instead of numbers. The calculated results are too large to display in integer form in the narrow columns of the top window, or with two decimal places in the bottom window. You can press ; and type **/GFI** to see the final sales figure at M1 in the bottom window. It should be 1712



Summary

We have covered a lot of ground in this lesson. The features and commands you have seen here can be mastered more easily than you might think. While you experiment, remember the following key points:

1. No matter what you type at the keyboard, you cannot hurt either the computer or the VisiCalc program. Moreover, it's fairly difficult to destroy your own work on the electronic sheet, particularly if you watch the prompt line for keystroke-by-keystroke feedback and save the sheet periodically on diskette.

2. Throughout this lesson, we have introduced only four new commands. They are:

   The **REPLICATE** command **/R** which gives you a shortcut way of writing similar formulas in adjacent rows and columns.

   The **GLOBAL** command **/G** which lets you change the column width (**/GC**) and the way numbers are formatted (**/GF**)

   The **TITLES** command **/T** which lets you fix rows or columns of alphabetic titles in place as part of the top or left hand "border".

   The **WINDOW** command **/W** which lets you split the screen, either horizontally or vertically, into two independently scrollable windows.

   Much of the power of the VisiCalc program is due to the simple and highly consistent way in which these commands interact with one another. In almost any context, a command will do what you would logically expect.

3. Aside from the REPLICATE command, which basically saves you time as you write formulas, all of the commands discussed in this lesson affect only the *appearance* of your work on the screen (generally in an effort to compensate for a small screen size). Nothing you might do with the /G, /T or /W commands can affect the labels, numbers or formulas actually written on the sheet. When in doubt, you can always type **/W1/TN /GFG/GC9Ⓔ** to return everything to normal.

Armed with these assurances, you should be ready to *experiment.* Clear the screen and try out these commands with a problem of your own. If you don't understand something, go back through this lesson to see what you might have missed, and check Part III of this manual (the VisiCalc Command Reference) and the reference card for more complete explanations. You now know enough about the VisiCalc program to begin to use it really effectively. Good luck!

Postscript: Memory and the Electronic Sheet

As you may know, your computer contains two kinds of fast semiconductor memory: RAM and ROM. ROM or "read only memory" is manufactured with a fixed pattern of data or program instructions stored in it. There is only a small amount of ROM for startup. RAM or "random access memory" is made in such a way that it retains data that is put into it only as long as the memory's electric power is left on or until new data is put into the memory in place of the old data. Data or program instructions are put into RAM memory either by typing at the keyboard or by loading prerecorded programs or data from diskette.

The VisiCalc program is loaded into RAM memory from diskette, and it, together with some RAM that the computer itself requires, uses some of the computer's memory. The remainder of RAM memory is devoted to the VisiCalc electronic sheet. The VisiCalc program manages this area of memory automatically. You never have to concern yourself with the details of how the electronic sheet is maintained. But you will find it useful to have a general idea of how the sheet works as you begin using the VisiCalc program to its fullest capacity.

Essentially, the VisiCalc program reconfigures the electronic sheet *dynamically*. That is, the VisiCalc program expands the size and shape of the sheet as you use it. You actually start with a 1 by 1 sheet, starting and ending at position A1. Although you can move the cursor to any position up to BK254, no memory is actually allocated for the sheet until you write something on it.

Then the sheet grows into a rectangle just large enough to include the rightmost and bottommost positions in which something is written. If you type a long message or formula at a particular entry position, the VisiCalc program will reserve additional bytes of memory for the position, but all of the other positions on the sheet will remain just large enough to hold the information which you have written in them. This is called "dynamic memory allocation".

The VisiCalc program continually displays the amount of memory available to you through the **memory indicator**. The memory indicator is the two-digit number located in the right corner of the prompt line just below the letter C. (See Lesson Three, Order of Recalculation for an explanation of the letter C.) This number is the amount of memory in kilobytes still available for additional entries on the sheet. The memory indicator will vary as you write information on the sheet or use commands, such as REPLICATE, to write information. The photos of the screen in this manual may show a different value for the memory indicator.

As you move further downwards and to the right, writing labels, numbers and formulas, this process continues, until the available memory is exhausted. If you finally exhaust all of the available memory, the VisiCalc program will replace the memory indicator with 0M. Your screen has expanded as far as the memory will allow. You can do some manipulation of the sheet but it is a good idea to reorganize and delete some of your worksheet or to start a new one. See part III, the DELETE Command and the section below on shrinking the sheet.

To sum up, the VisiCalc program manages memory efficiently and automatically. Because memory is dynamically allocated and the sheet is reconfigured to suit your needs, you can work ahead of yourself in terms of memory. Note, however, that you certainly shouldn't be wasteful with it. Aside from noticing from time to time how much space is still available for you to write in more entry positions, you need not concern yourself with problems of memory management.

Shrinking the Sheet

One thing the VisiCalc program does not do automatically, however, is to "shrink" the sheet. Suppose that you have written on various portions of the sheet, causing it to grow to a size of 100 by 100 positions. Later, having finished with this information, you might have erased or blanked out the entry positions near the right and bottom edges of the 100 by 100 sheet.

However, the sheet will remain configured as 100 by 100, although each empty position will require approximately 2 bytes. If you begin using additional memory by writing lengthy labels and formulas in other positions, you may want to shrink the sheet to the smallest possible configuration for the information still written on it.

To do this, you must save the sheet on diskette with the **/SS** command (as discussed in Lesson One), clear the sheet with **/CY**, and reload the saved sheet with the **/SL** command (as discussed in Lesson Two). As the disk file is loaded, the VisiCalc program will enter on the sheet only those labels, numbers and formulas actually saved, and in the process the sheet will grow from a 1 by 1 configuration to just the size you need for the information remaining. This will make more memory space available for additional labels and formulas.

Lesson Three

Lesson Three

In Lessons One and Two, we used several examples to illustrate both the simplicity and the power inherent in the VisiCalc program's concepts and features. Although each individual VisiCalc command is quite simple, the various commands can be used in combination with each other to achieve a wide range of useful effects. As with any tool, there is a skill involved in using the VisiCalc program effectively, and you will gain this skill through experience. In this lesson, we will begin to illustrate some of the techniques you can use to get the most out of the VisiCalc commands. We will also introduce more VisiCalc commands and features in this lesson.

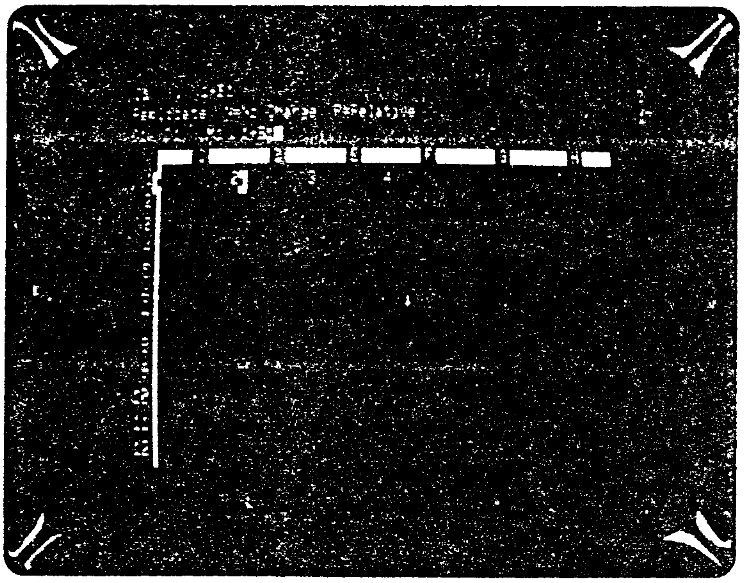
Let's begin with a clean slate. Load the VisiCalc program into your computer as described in the section entitled "Loading the VisiCalc Program," or, if you already have the program running, clear the sheet by typing **/CY**

In this lesson we'll outline a household budget application, with suggestions as to how you might adapt it for your own use. To prepare a budget, we'll first project our income for the next twelve months. We'll also project various necessary expenses such as food, rent or mortgage, telephone, etc., as well as semiannual expenses such as car insurance. Then we'll use the VisiCalc program to find out how much of our income is left for leisure and for savings and what percentage of our income is going for each category of expense. Finally, we'll consider various enhancements such as calculating the interest on our savings account.

Typeahead

First, we'll lay out twelve months or periods across the sheet. Type the word **PERIOD** and press the ➧ key to move on to position B1. Now, as fast as you can, type the following keys: **1➧2➧3➧4➧** and watch what happens on the screen. Were you typing faster than the VisiCalc program could move the cursor and write the numbers on the sheet? If so, you noticed a feature called **typeahead**: the VisiCalc program remembers the keystrokes you type, no matter how fast you go, and it catches up with you as soon as it can. (If you are wondering why the VisiCalc program was so slow in the first place, read the section in Lesson Two entitled "Memory and the Electronic Sheet" and you'll realize that the sheet was actually "growing" as you typed.)

Before going on to type the numbers 5 through 12, let's ask ourselves: Is there a better way? Let's use the REPLICATE command and let the VisiCalc program **calculate** the numbers 1 through 12. Use **>A1**Ⓔ▸▸ to move the cursor back to C1, where the number 2 is written. Can you obtain the number 2 from a formula? It's simple: Type **1+◂**Ⓔ The entry contents line should read C1 (V) 1+B1 and the value 2, now the result of 1+B1 or 1+1, should still be present at C1. Now, let's replicate. Type **/R**Ⓔ The prompt line reads Replicate: Target range and the edit line reads C1 . . . C1: followed by the edit cursor. Press ▸ to obtain the starting position, D1; then press **.** and finally, hold down the ▸ key to move the cursor to column M, which will be period 12. (If you overshoot, back up with the ◂ key.) The edit line should read C1 . . . C1: D1 . . . M1 Now press Ⓔ The prompt line reads Replicate: N=No Change, R=Relative and the edit line reads C1: D1 . . . M1: 1+B1 with the edit cursor after B1 as in the photo below.



Press **R** to make the coordinate relative: This will give us 1+C1, 1+D1, etc. The prompt and edit lines should go blank. Move the cursor out to column M to check your work: Position M1 should show the number 12.

Replicating Numbers and Labels

Type the following characters, ending with the Ⓔ key as shown:

>A2Ⓔ
INCOME▶1800Ⓔ

We'll assume that $1800 is your monthly "take-home pay" after taxes and other deductions. Now, let's fill in the figure 1800 for all twelve months. Press **/RⒺ** Can you replicate a single number as well as a formula? Of course: A number is actually the simplest case of a formula. For the target range, type **C2.M2Ⓔ** You aren't asked whether the new formula is relative or not, because there are no coordinates in the "formula" 1800. The number 1800 should now appear in all twelve columns, in positions B2 through M2.

Next, we'll draw a line across the sheet. Move the cursor with **>A3Ⓔ** and then type **/–** The prompt line reads Label: Repeating appears on the edit line. Whatever character or characters we type next will be repeated to fill the entry position A3. Type **–** followed by Ⓔ You should now have a line of nine hyphens at A3. Is this any different from simply typing the hyphens manually? Type **/GC12Ⓔ** As you can see, the repeating label expands to fill the widened entry position. Now, go back to **/GC9Ⓔ**

The ever-useful REPLICATE command will also replicate labels. Type **/RⒺ**. For the target, type **B3.M3Ⓔ**. It's that simple. You should now have a line of hyphens extending all the way to column M.

Using Formulas for Flexibility

Before we go any further, let's think about what we've done. To save ourselves the trouble of typing the number 1800 twelve times, we replicated this number. That's fine as far as it goes, but is it the best way to handle our income? It would be better if we could change the income figure for all twelve months by simply typing a new figure for the first month and taking advantage of the VisiCalc recalculation feature. Let's replicate a formula instead of a number. Type:

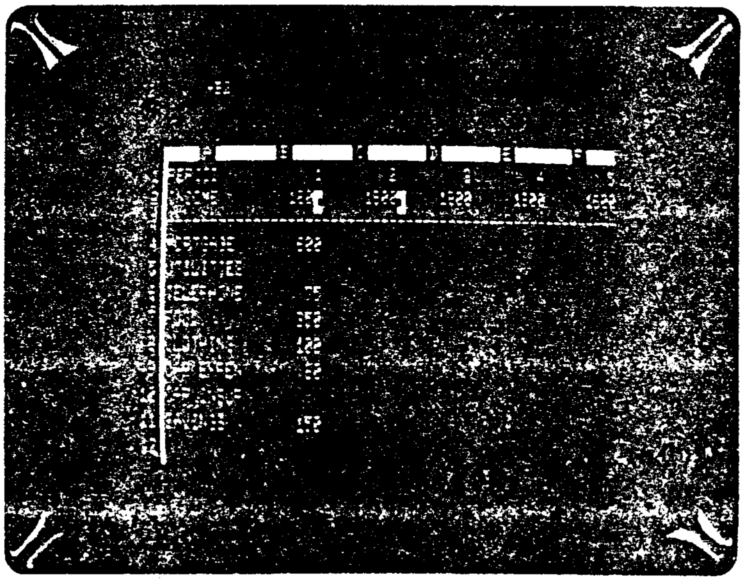
>C2Ⓔ
+B2Ⓔ

We have defined the second month's income to be the same as the income for the first month. Next, let's replicate: Type **/RⒺ** The target range is **D2.M2Ⓔ** Now the prompt line reads Replicate: N=No Change, R=Relative. Do we want the same formula, +B2, in all of the remaining positions, or would we prefer +B2, +C2, +D2, etc.? Either way we can change the income for all twelve months simply by typing a new number at B2. Think further: What if we should get a raise in the sixth month? If the formulas refer to the previous month, we can simply type a new number in month 6 and "propagate" the change through months 7–12. Let's try it. Type **R** to make the coordinate B2 relative. When the replicate command has finished, use the ▶ key to move to month 6 (position G2). Now type **2000Ⓔ** Press ▶ a few more times to verify that each succeeding month's income has changed to 2000. Were you able to foresee the way in which the change would be propagated? If you weren't sure, move the cursor over all twelve income figures and imagine what would have happened if all of the formulas were +B2.

Our next task is to list our expense categories and estimate monthly amounts for each category. (Some expenses will vary from month to month, and other expenses will occur perhaps only every six months. We will leave these blank for the moment.) You can either type the following exactly as shown, or you can use the arrow keys to move the cursor and save yourself some keystrokes. (Hint: To take full advantage of the arrow keys, type all the alphabetic labels first.)

```
>A4Ⓔ
MORTGAGE♦600♦
>A5Ⓔ
UTILITIES♦
>A6Ⓔ
TELEPHONE♦75♦
>A7Ⓔ
FOOD♦350♦
>A8Ⓔ
CLOTHING♦100♦
>A9Ⓔ
CAR EXPENSE♦80♦
>A10Ⓔ
CAR INSURANCE♦
>A11Ⓔ
SAVINGS♦150♦
>C2Ⓔ
```

At this point your screen should look like the screen photo below:



Next, we would like to replicate the monthly expense figures in column B across for the remaining eleven months. Remember our discussion of the merits of replicating a number versus a formula for our monthly income? To give ourselves maximum flexibility, we should also replicate formulas for the monthly expenses. At C4 we want the formula +B4; at C6 we want the formula +B6; at C7 we want +B7; and so on. (We'll fill in figures for UTILITIES and CAR INSURANCE later.) These formulas are so similar to each other and to the income formula +B2 that it's tempting to look for a shortcut way of typing them. Once again, the REPLICATE command comes to our aid. This time, we'll replicate a formula down a column instead of across a row.

Replicating Down a Column

Make sure that the cursor is at C2. The entry contents line reads C2 (V) +B2 In a relative sense, C2 is to B2 as C4 is to B4, and so on: We want to treat the coordinate B2 as **relative.** Type **/R**Ⓔ The prompt line reads Replicate: Target range and on the edit line is C2 . . . C2: followed by the edit cursor. Press the ↔ key twice. Now the edit line reads C2 . . . C2:C4 Type a period. The cursor jumps back to C2; the VisiCalc program acknowledges that the target range will start at C4. Next, press the ↔ key nine times (or hold it down to utilize the auto repeat ability) to reach position C11, opposite the figure for SAVINGS. The edit line now reads C2 . . . C2:C4 . . . C11 so the target range will be C4 through C11. Press Ⓔ The cursor jumps back to C2, and the prompt line reads Replicate: N=No Change, R=Relative. The edit line reads C2:C4 . . . C11: +B2 with the edit cursor following the coordinate B2. We want this coordinate to be relative: Press **R** A column of numbers, from 600 to 150, appears in column C. Use the ↔ key to move the cursor downwards, pausing to look at the formulas we've replicated. We have what we wanted: At C4 is +B4, at C6 is +B6, and so on. We also have formulas at C5 and C10, but we can easily eliminate them. You can use ↔ and **/B** to do this, or you can type exactly the following:

```
>C5Ⓔ
/B◆
>C10Ⓔ
/B◆
>C4Ⓔ
```

Replicating a Column Several Times

We now have the formulas we want for each expense category. The next step is to replicate these formulas across the rows through month 12. Do you remember how we replicated a source range of formulas, for both cost of goods sold and gross profit, across the rows in Lesson Two? We can do the same thing here. Press **/R** The prompt line reads Replicate: Source range or ENTER and on the edit line is C4 followed by the edit cursor. Press the ➡ key seven times to move the cursor down to C11. The edit line reads C4 . . . C11 followed by the edit cursor. Now press Ⓔ The cursor jumps back to C4, and the prompt line asks us for a Target range. Type **D4.M4Ⓔ** We have asked the VisiCalc program to replicate the formula at C4 into positions D4 through M4; the formula at C5 into positions D5 through M5; the formula at C6 into positions D6 through M6; and so on, through the formula at C11. Notice that we gave only the first coordinate in each column in the target range. Now the prompt line reads Replicate: N=No Change, R=Relative. On the edit line is C4: D4 . . . M4: +B4 with the edit cursor following B4. This is the formula for the first expense, mortgage. As was the case for B2 (income), B4 should be relative. Press **R** and watch the screen. Several things happen:

1. The cursor disappears.
2. The number 600 appears at positions D4, E4 and F4.
3. The edit line now reads C6: D6 . . . M6: +B6.

The VisiCalc program has finished replicating the formulas +B4, +C4, +D4, etc., in row 4, and has also replicated the blank entry at B5 into C5, D5, E5, etc. Next, the VisiCalc program wants to know how to handle the formula +B6 on row 6. All of these formulas will be relative; press **R** five more times. You have written a total of 80 numbers and formulas on the electronic sheet, with the aid of the REPLICATE command.

Think about the technique we used to replicate the expense formulas: starting with the prototype formula +B2 at position C2, we created six more prototype formulas by replicating down a column: +B4 at C4, +B6 at C6, etc. Then, we used these formulas as our source range to replicate similar formulas across on rows 4 through 11. Each of the resulting monthly expenses can be changed for all twelve months simply by typing a new number for the first month. For example, type **>B8Ⓔ** and change the number there to **120Ⓔ** Your clothing budget is raised to $120 for all twelve months.

To complete our projection of expenses, we'll fill in figures for those expenses which cannot be replicated across because they vary from month to month. Our utilities bill will vary depending on the season and the need for heating or air conditioning. Our car insurance premiums are due every six months; we'll pay a premium in month 1 and month 7. Type the following exactly as shown:

```
>B5Ⓔ
140➡140➡80➡80➡40➡40➡85➡85➡50➡50➡100➡140➡
>B10Ⓔ
160➡
>H10Ⓔ
160➡
```

We do not have to fill in zeros for the other ten months in the row for car insurance, because the VisiCalc program treats any blank entry as equivalent to zero. (In fact, any entry containing an alphabetic label will have a "value" of zero if it is referenced elsewhere in a formula.)

Now is a good time to save your work on diskette, if you have not already done so. Make sure that your disk drive's ACTIVE light is off. You should have a storage diskette in drive 0. If not, insert a storage diskette into drive 0 and type /SS. In response to the prompt Storage: File for Saving, type **LESSON3A/VCⒺ**. You should also, at this time, make a backup copy of this file on another diskette which you do not use except for storing the backup files. If you should decide to end this session at this point, **be sure to remove any diskettes from the disk drives before you turn off the TRS-80 Model III.**

Fixing Titles in Both Directions

We again have a situation where, if we scroll the window to look at the later months, we will lose the descriptive titles INCOME, MORTGAGE, etc. Let's fix them in place. This time, we'll create a border of titles along both the left and top edge of the sheet.

Move the cursor with **>A3Ⓔ** Now type /T The prompt line reads Titles: H V B N. (The meanings of the keystrokes H, V, B and N are described in Lesson Two and in Part III, The VisiCalc Command Reference.) Press B to fix titles in **both** directions. The position of the cursor has a dual significance for this command. If you type /TV to fix titles vertically, the VisiCalc program will fix in place the **column** in which the cursor rests, and all columns currently on the screen to the left of the cursor. If you type /TH to fix titles horizontally, the VisiCalc program will fix in place the **row** in which the cursor rests, and all rows currently on the screen above the cursor. Typing /TB is equivalent to typing both /TV and /TH, so the exact position of the cursor is important. In this case, we have created a border consisting of column A along the left edge of the screen window, and another border consisting of rows 1, 2 and 3 along the top edge of the window. To check this, hold down the ↓ key to scroll the window downwards past row 12. Notice how rows 1, 2 and 3 remain in place. Now, type >B4Ⓔ and press the → key to see how the window scrolls horizontally, leaving column A in place. Continue pressing the → key until columns A, M, N and O are in the window. We'll use columns N and O to obtain totals and percentages for our income and expenses.

The Built-in Function @SUM

Type the following:

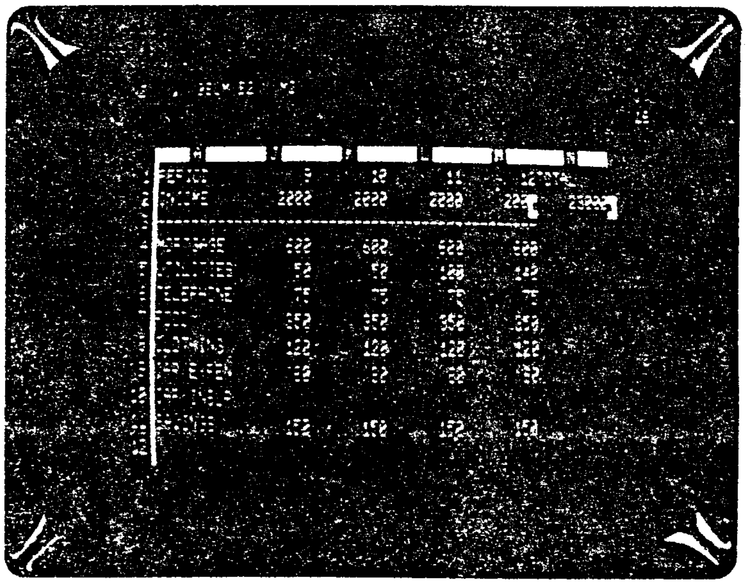
>N1Ⓔ
TOTAL→

The cursor should be at N2. How can we find our total income for twelve months? You could always type

+B2+C2+D2+E2+F2+G2+H2+I2+J2+K2+L2+M2

but there's a simpler way. Press @ watching the prompt line as you do so. As soon as you type the @ symbol, the prompt line says Value. Now type **SUM(** and the edit line reads @SUM(followed by the edit cursor.

Our next step is to indicate what numbers we want to sum up. Hold down the ← until you begin "bumping into" the left hand border. The cursor should be at B2, and the edit line should read @SUM(B2 followed by the edit cursor. Press **.** The cursor jumps back to N2, where we started, and the edit line now reads @SUM(B2 . . . As you can see, we are specifying a range of entries, just as we have done several times for the REPLICATE command. To finish this range, press ←)Ⓔ The entry contents line now reads N2 (V) @SUM(B2 . . . M2), and under the cursor highlight is the number 23000, our total income for the year.



The @ symbol is used to begin the name of a VisiCalc built-in function. When you began the entry at N2 with the symbol @, the VisiCalc program immediately knew that:

1. The entry was going to be a numeric VALUE.
2. The next few letters you typed would be the name of a built-in function.

Each built-in function, such as @SUM, performs some sort of calculation on the list of values given to it, and yields a numeric result. Other examples of built-in functions are **@MIN, @MAX** and **@AVERAGE**; see the Reference Card and Part III, The VisiCalc Command Reference, for a complete description of all the functions. Functions such as @SUM will operate over a range such as @SUM(B2 . . . M2); a list of particular values such as @SUM(B2, B7, C3, D8); or a list of ranges, values, numbers or formulas, such as @SUM(B2 . . . B7, C3 . . . C6, 25, D8, 4*C8). And the numeric result of the function can be used wherever an ordinary number could be used: for example, in a function or another formula.

For our personal budget, we would like to obtain totals for each of our expense categories, just as we did for income. Once again, a replication is called for. With the highlight on N2, which is our prototype formula, press **/RⒺ** The target range is **N4.N11Ⓔ** Now the prompt line reads Replicate: N = No Change, R = Relative. The edit line reads N2: N4 . . . N11: @SUM(B2 with the edit cursor following B2. Think for a moment about what will happen if we make B2 . . . M2 relative. Since we are replicating down a column, the replicated formulas will be in positions N4, N5, N6, etc. We want N4 to be the sum of B4 . . . M4; N5 to be the sum of B5 . . . M5 and so on, which is what will happen if the replicated formulas are relative. Press **R** twice. In an instant, the expense totals, from 7200 to 1800, appear in column N.

NOTE that if you insert a new row or column into an @SUM range with the **INSERT** command, the formula automatically expands to include the new material. Two important rules apply to this procedure. First, *never* place the insertions on the first coordinate in the range. Second, *never* place the insertion on the coordinate of the @SUM formula. For example, if you have material in B5 thru B10 and the @SUM formula in B11, you cannot insert a row of new material at B5 or B11. Insertion at these positions places the information outside of the range. You can insert at B10.

Let's find out what percentage of our income is accounted for by each expense total. Type the following (the first line below uses the letter O):

>O1Ⓔ
PERCENT◆◆◆

What formula will give mortgage as a percentage of income? Type: **+N4/N2Ⓔ** We are dividing the mortgage total at N4, or 7200, by the income total at N2, or 23000. The result is .3130435, or approximately 31%.

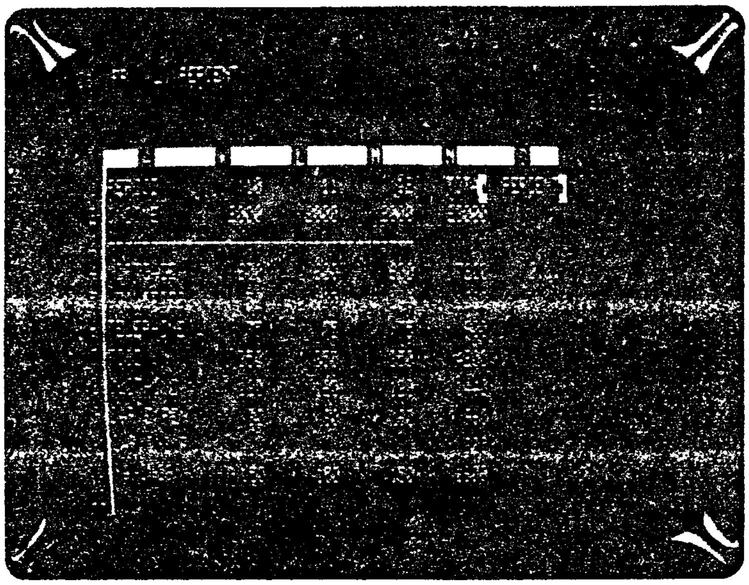
Formatting a Single Entry

The number .3130435 is unnecessarily messy. This is the general format (showing as many significant digits as the column width will allow), which the VisiCalc program uses in all cases unless we ask for something else. For our purposes, two decimal places should be enough. The global format command /GF$ which we used in Lesson Two displays everything to two decimal places. Type **/GF$** Our mortgage percentage at position O4 now appears as 0.31, but every other number on the screen is also displayed in dollars and cents form. Even the month number 12 at M1 appears as 12.00. This is still messy. Change the global format back to "general" by typing **/GFG** What we really want is to display just the number at O4 in dollars and cents form. With the cursor still at O4, type **/F$** That's it! Just as the command /GF$ affects everything in the screen window, the command **/F$** affects only the entry highlighted by the cursor.

Now that the percentage 0.31 is pretty, look at the label TOTAL at the top of column N. It is hard to read, because it's too close to the number 12 in column M, and it doesn't line up with the numbers below it. Move the cursor to the label TOTAL with **>N1Ⓔ** The VisiCalc program normally starts alphabetic labels from the left side of an entry position (the **general** format for labels, which currently applies **globally** to all entries including PERIOD, INCOME, etc.), whereas numbers have their last digits lined up on the right. To change the format of TOTAL, type **/F** The prompt line reads Format: D G I L R $ * The possible keystrokes are:

D	The format **defaults** to be the same as the global format. In other words, there is no special format for this entry.
G	Use the **general** format for this entry. You can use this, for example, to display a particular number with several decimal places even if the global format is I (integer).
I	Display this entry in **integer** format.
L	Start this entry (a number) at the **left** side of the entry position.
R	Make the last letter or digit of the entry line up at the **right** end of the position.
$	Display this entry in **dollars and cents** format.
*	Display this entry in **graph** format. For examples, see Part II, Lesson Four; Part III, The FORMAT Command; and the Pocket Reference.

For TOTAL, type **R** Now TOTAL is lined up with the column of figures below it. Now press ▶ to move the cursor to O1, and press **/FR** to "right-justify" the label PERCENT.



Replicating a Format Specification

Type **>O4**Ⓔ to move to the formula for mortgage expense as a percentage of income. We'll replicate this formula down column O to obtain percentages for all of the other expenses. Type **/R**Ⓔ The target range is **O5.O11**Ⓔ Now the prompt line reads Replicate: N=No Change, R=Relative and the edit line reads O4: O5 . . . O11: +N4 How should we handle the formula +N4/N2, which is mortgage as a percentage of income? For utilities at O5, we want +N5/N2. The first coordinate changes, but the second, N2 or income, does not. Press **R** to make N4 relative; then, with the edit cursor following N2 on the edit line, press **N** to indicate "no change." Now we have all of our expense percentages, from 0.31 for mortgage to 0.08 for savings. Notice that all of the percentages are displayed to two decimal places. Press ◆ a few times, looking at the entry contents line. At O5, for instance, the line reads O5 /F$ (V) +N5/N2 The format specification /F$ has been replicated along with the formula. In fact, it's possible to replicate a format specification even if the entry being replicated is blank! We'll use this trick a little later.

Using Replicate To Copy a Row or Column

Type >A12Ⓔ Now we'll draw a line under our list of expenses; then calculate how much money we have left for leisure. You already know one relatively easy way to draw a line of hyphens, by using the /− command at A12 and replicating the hyphens across. But let's try using the replicate command a little differently. Type **/R** The prompt line reads Replicate: Source range or ENTER and on the edit line is the cursor coordinate, A12, followed by the edit cursor. Now press **CLEAR** The A12 disappears, leaving only the edit cursor on the edit line. Let's type a new source range: **A3.M3Ⓔ** As usual, the prompt line reads Replicate: Target range. Type **A12.A12** Just what are we doing? The source range, A3 . . . M3, is the line of hyphens already written on the sheet. We're asking the VisiCalc program to replicate the entry at A3 into position A12; the entry at B3 into position B12; and so on. Press Ⓔ There's your line. (Incidentally, it would have been sufficient to type **A12Ⓔ** for the target range; the VisiCalc program will take this to mean A12 . . . A12.)

Finally, let's add LEISURE to our budget. The money we have available for leisure will simply be our income minus the sum of our expenses. Type the following:

```
>A13Ⓔ
LEISURE▸+B2−@SUM(B4.B11)Ⓔ
```

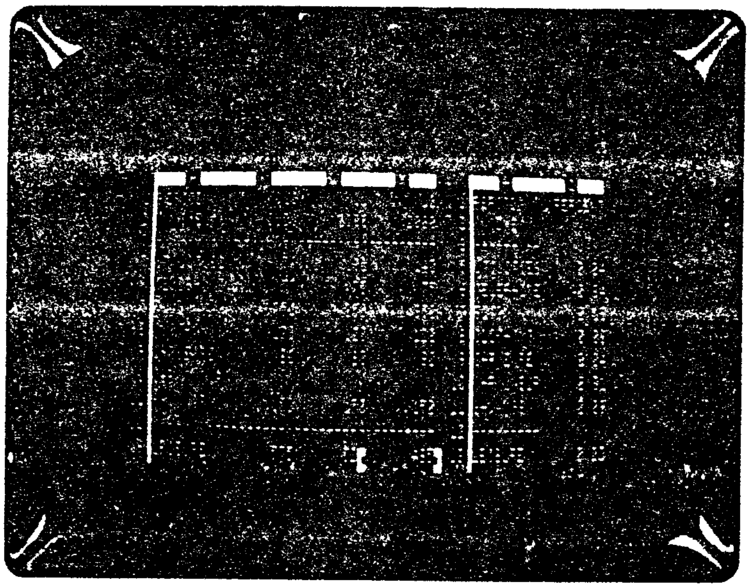
The figure for leisure, 125, appears at B13. Then replicate this formula across with **/RⒺ C13.M13Ⓔ** and press **R** three times to make the formula coordinates relative. Now start pressing the ▸ key to scroll the window rightwards, and examine the encouraging results the VisiCalc program has calculated for us. Our leisure money starts small, at $125 in the first month, but it increases fairly steadily thereafter. After the first month, we don't have a car insurance premium, and in months 3, 4 and 5 our utilities bill goes down. In month 6, we get our raise, which increases our income and leisure money. Continue pressing ▸ until you have brought columns M, N and O into view, showing TOTAL and PERCENT, and then press ◂ to leave the cursor at N13. To obtain a total and percentage of income for leisure, we can use the same method we used earlier to copy the line of hyphens. Press the following keys, and watch what happens on the prompt and edit lines (here in the REPLICATE command, we may press : in place of Ⓔ where : is shown).

```
/R CLEAR N11.O11:N13ⒺRRRN
```

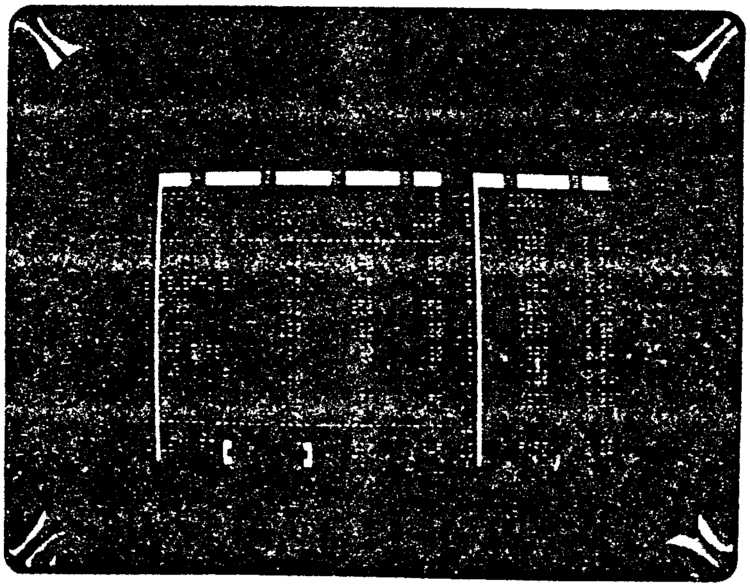
You should have a total of 5150 and a percentage of 0.22 for LEISURE.

Changing Windows and Titles

As we found in Lesson Two, the screen is too small to display both our starting expense figures in month 1 and our calculated totals and percentages in columns N and O. We can solve this problem by splitting the screen into two windows. With the cursor still at N13, press /WV Your screen should look like the photo below. The titles in column A and rows 1–3 are fixed in place in both windows.



Press ; to jump the cursor over to the right window, bringing columns A and N into view. Then press /TN to eliminate the fixed titles from this window. The title or "border" column A disappears, exposing column M. Finally, press → once to bring the TOTAL and PERCENT columns N and O into view, and press ; to jump the cursor back to the left window. Press and hold down ← to scroll this window back to the first month. (Notice that you bump into column A. The fixed titles are still in effect in this window.) Your screen should look like the screen photo below.



Let's try changing one of our expenses to see how the VisiCalc program recalculates the expense totals and income percentages. Type >B9 Ⓔ 100 Ⓔ. Your available leisure money decreases by $20 each month; the car expense total goes from $960, or 4% of your income, to $1200, or 5% of income; and your leisure total for the year goes from $5150, or 22% of income, to $4910 or 21%.

The @NA and @ERROR Functions

Press ➡ to move to position B10. Let's suppose that you were about to change your car insurance policy, and the new premium was not yet known. How would you deal with this in your budget? The VisiCalc program has a special way: Replace the number 160 at B10 by typing @NAⒺ and watch the screen. As you might have guessed, @NA stands for "Not Available." When you write this function at an entry position, that entry takes on the special value NA. When the VisiCalc program recalculates, any formula which refers to an entry containing NA will itself have a value of NA. In this case, your car insurance premium for month 1 at B10 became Not Available. Because of this, there was no way to calculate your leisure money (income minus the sum of expenses) for month 1, and so position B13 (for leisure) became Not Available. Moreover, your total car insurance expense for the year could not be calculated and also became NA, which meant that the corresponding percentage of income became NA. And, since one month's leisure expense was NA, the total was NA, and so was leisure as a percentage of income. Change B10 back to **160**Ⓔ and all of the calculated values will be restored.

Here's a related issue: Suppose that, instead of typing a number, you were trying to calculate your own car insurance premium as the insurance company does, but you made a mistake and tried to divide by zero. Type **1/0**Ⓔ As the screen shows, when the VisiCalc program evaluates the formula 1/0, the result is the special value ERROR. Like NA, the value ERROR "propagates:" Any formula which refers to an entry with the value ERROR will itself have the value ERROR. There are several ways to obtain a value of ERROR: dividing by zero, taking the logarithm of a negative number, calculating a value that is simply too large for the computer to represent, etc. Too deep a nesting of parentheses in a formula can produce ERROR. A syntax error such as a wrong argument in a function can also produce ERROR. You can also deliberately obtain the value ERROR, just as you did for @NA, by typing the function name @ERROR. @ERROR can be used to check out lookup tables. For now, change the value at B10 back to **160**Ⓔ so that the totals and percentages can be calculated.

To make sure that we can continue from this point if something goes wrong, let's save the sheet on diskette again. You can save a sheet with ERROR in it. The storage diskette which you used earlier in this lesson should still be in place in your disk drive. Type /**SS** and, in response to the prompt Storage: File for Saving, press the ⬍ key. The disk drive should run, and a file name should appear on the edit line. Continue pressing the ⬍ key, if necessary, until you have the name LESSON3A/VC:0 on the edit line. Then pres the **CLEAR** six times and type **B/VC:0**Ⓔ watching the edit line as you type the characters. You are saving the sheet under the revised name LESSON3B/VC:0. Once you've started, all it takes is a few keystrokes to protect yourself.

Now, press ; to jump the cursor into the right hand window. There are no fixed titles in effect in this window (the VisiCalc program sometimes automatically adjusts the column widths to make room for the extra vertical border of row numbers which runs down the center of the screen.) Press **/W1** to leave the narrowed columns on the screen. When you type /W1 to return to one screen window, the global column, format, and title settings for the full screen are obtained from the window where the cursor lies at the time you type the /W1 command. If you had typed /W1 with the cursor in the left hand window, the screen would now have fixed titles and columns nine characters wide. For now, type **/GC9Ⓔ** and **>A7Ⓔ** to adjust the column widths and scroll the screen back to the left edge of the sheet. The cursor should highlight the label FOOD.

The Insert and Delete Commands

Let's say you decide to take on a life insurance policy which has monthly premiums of $115, and you want to incorporate this expense into your budget. If you were working on an ordinary sheet of paper with the same arrangement of figures which we have on the screen, you'd have to erase something, or write in tiny letters or off to the side to make room. But the VisiCalc electronic sheet is more flexible. Type **/IR** for "insert row" and watch the screen. Everything at or below the cursor moves down to make room, leaving you with a blank line at row 7 where you can enter the life insurance figures.

Notice that, for example, the amounts for savings, which used to be on row 11, are now on row 12. This may remind you of the formula you wrote for leisure, which was income minus the sum of expenses: +B2−@SUM(B4 . . . B11). Now savings at B12 is outside this range. But the figures for leisure haven't changed. Type **>B14Ⓔ** and look at the entry contents line. The formula has changed to be +B2−@SUM(B4 . . . B12)! Whenever you insert (or delete, or move) a row or column, the VisiCalc program automatically adjusts all of the formulas on the sheet so that they refer to the same entry positions as before, even though the coordinates have changed.

Now type the following to fill in the life insurance expense figures. Since all we have is a blank line at row 7, we will also have to add the formulas to calculate total and percent for this new expense.

```
>A7Ⓔ
LIFE INS▶115▶+◀Ⓔ
/RⒺD7.M7:R
>N6Ⓔ
/R▶:N7:RRRN
>A10Ⓔ
```

Notice that our leisure money has decreased by the amount of the life insurance premiums each month. Position B14 is now −10, meaning that we're overspending our income in month 1. Perhaps we should ride the bus to work, and in that way all but eliminate our car expense. Type **>B10Ⓔ** and then press **/D** watching the screen as you do so. The prompt line reads `Delete: R C` (With the cursor at B10, you could delete row 10 by pressing R, or column B by pressing C.) Press **R** The title and figures for car expense disappear completely from the screen, and everything below the cursor moves up one row to take up the slack. Car insurance is now at row 10, and savings is back at row 11. Our leisure money has increased, back to $90, for example, at position B13. (If you were to check the formula at B13, you'd find that it has been adjusted back to +B2−@SUM(B4 . . . B11).)

You can also insert columns. Say, for example, that you wanted to obtain six month totals for your income and expenses. Type **>H4Ⓔ** The formula there is +G4, and you know the formula at I4 is `+H4`. Now press **/IC** A new, blank column appears in place of column H. (Notice that the insert command always inserts a row or column before, meaning "closer to row 1 or column A than," the row or column where the cursor lies.) Press **➧** to bring into view the old column H, now relettered column I. The formula at I4 is `+G4`, so the propagation of the income and expense amounts skips over the newly created blank column. Press **➧** twice more, and notice that the next formulas are `+I4` and `+J4`. Now move back with **>H4Ⓔ** and type **/DC** Now the blank column has been deleted, all of the other columns to the right have moved back to eliminate the empty space, and everything has returned to normal. (If you've made any mistakes in this section, you can clear the screen, reload the diskette file name LESSON3B/VC:0, and try each step again.)

To test your understanding of the insert and replicate commands, try the following on your own, then compare your approach with the instructions below: Insert the names of the months, such as JAN, FEB, MAR, etc. just below the month numbers 1 through 12.

Have you succeeded? Here's one way to do it. Type exactly the following:

```
>A2Ⓔ
/IRMONTH➧/FR/R:C2.M2Ⓔ (This replicates the format /FR)
JAN➧FEB➧MAR➧APR➧MAY➧JUN➧JUL➧AUG➧
SEP➧OCT➧NOV➧DECⒺ
>A2Ⓔ
```

Calculating Interest on a Savings Account

According to our present budget, we are setting aside $150 each month for savings. Part of the motivation for doing this, of course, is that we would like to earn interest on this money. Let's use the VisiCalc program to project the interest and the accumulated balance we would have if we put this money into a savings or other investment account.

We'll assume that interest on a savings account is paid at the rate of 5% per year, compounded monthly. But, to give ourselves flexibility, we'll write this interest rate into a separate position on the sheet, so that we can change it later and let the VisiCalc program recalculate the interest and accumulated balance. Type **>A15Ⓔ SAV ACCT➡.05Ⓔ**

On the first day of each month, our account will be credited with interest for the balance in the account during the previous month, and then we'll deposit our monthly $150. Type the following:

```
>A17Ⓔ
/F$100➡
>A16Ⓔ
INTEREST➡
```

The $100 at A17 is the previous balance in the savings account before our budget begins. The interest paid for one month will be one twelfth of the yearly rate (e.g., .05 divided by 12) times this previous balance. Type:

```
/F$+B15/12*A17Ⓔ
```

The calculated result of this formula should be 0.42, shown at B16. Does this figure make sense? A year's simple interest at 5% on $100 would be $5.00. One twelfth of this would be $5.00/12, or $0.41666 (which rounds to $0.42).

Press ⬇ to move to B17. Our new savings account balance will be the previous month's balance, plus the interest, plus the savings deposit for this month. Type:

```
/F$+A17+⬆+B12Ⓔ
```

(If you like, you can point with the cursor to fill in all three coordinates in this formula.) The result, as expected, is $250.42.

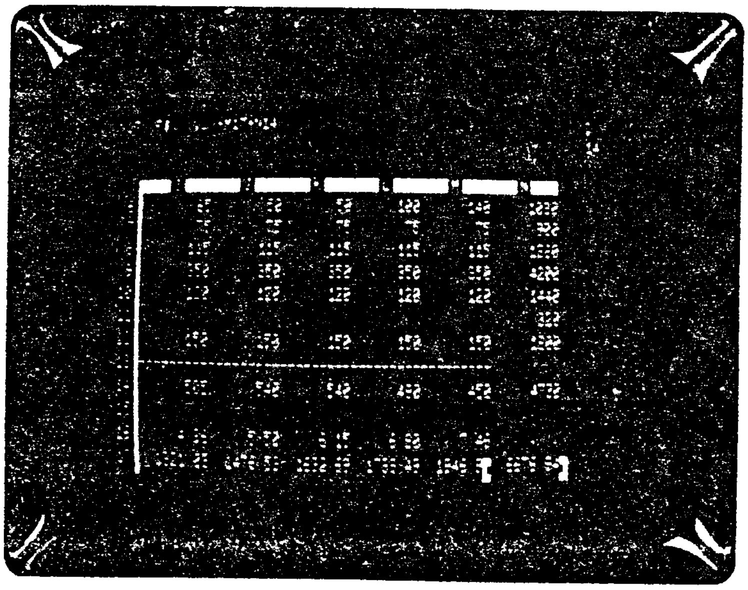
Now, we can replicate both the interest and account balance formulas across for twelve months. Type:

```
>B16Ⓔ
/R←:C16.M16:NRRRR
```

As you type the N and R's to determine whether each coordinate should be unchanged or relative, think about what each one means. The interest rate at B15 is clearly N. The R for A17 means "use the value of the entry one position down and to the left of each copy of the formula," and so on for the other coordinates. Press → to examine the calculated results. The interest paid each month increases, since both our monthly deposits and the accumulated interest are added into the balance on which the interest is based. Continue scrolling the window rightward until column N comes into view, and then type the following:

```
>N17Ⓔ
/F$+M17+N14Ⓔ
```

This is the combined total of our savings and leisure money, or our "discretionary income." It should currently be $6676.94.



The Move Command

Type >**A11**Ⓔ The cursor should highlight the label CAR INSURANCE. Paying that insurance premium in month 1 is taking a big bite out of our available leisure money in month 1 (which is $90, as opposed to $250 in month 2). Perhaps we can pay the insurance premium by taking money out of the savings account.

On an ordinary sheet of paper, more erasures and writing in the margins would be necessary. Not so on the VisiCalc electronic sheet. Type **/M** The prompt line reads Move: From . . . To and on the edit line is A11 the cursor coordinate, followed by the edit cursor. Now press ↓ The cursor moves down to highlight SAVINGS, and the edit line reads A11 . . . A12 Press the ↓ key four more times, watching the edit line change from A11 . . . A12 to A11 . . . A16 much as it did for the @SUM function and the REPLICATE command. The cursor highlights INTEREST at A16. Now press Ⓔ and watch the screen. The following things happen:

1. The entire row for CAR INSURANCE moves down from row 11, reappearing just above INTEREST at row 16.
2. The rows for SAVINGS, LEISURE and SAV ACCT move up, taking up the space vacated by the old CAR INSURANCE row, and making a new space just above INTEREST for the new CAR INSURANCE row.
3. The cursor remains where it was when we started the /M command: at A11, which is now SAVINGS.

Now look at the figure for leisure in month 1. It has increased from 90 to 250. The car insurance premium has been taken out of the sum of expenses used to calculate leisure (the formula at B13 now reads +B3−@SUM(B5 . . . B11)). Hence, the car insurance expense is currently unaccounted for. We must revise the formulas in row 17 to take the car insurance premium out of the savings account balance. Type >**B17**Ⓔ and look at the entry contents line. The formula reads +A17+B16+B11 meaning the previous account balance, plus a month's interest on that balance, plus this month's savings deposit. To this we must add "minus the car insurance premium (if any)." Type the following:

+A17+B16+B11−↑↑Ⓔ
/R:C17.M17:RRRR

This will replicate the new account balance formulas +A17+B16+B11−B15, +B17+C16+C11−C15, etc., across the row.

Now type >**N17**Ⓔ to examine our total discretionary income: It has declined from 6676.94, before we took the insurance premiums out of savings, to 6666.10. We are now saving less and spending more on leisure, so we have lost about $10 interest on our savings account. (Notice that our leisure total at N13 has increased from 4730 to 5050.) Perhaps we should save a little more each month to replenish the funds taken out to pay the insurance premiums.

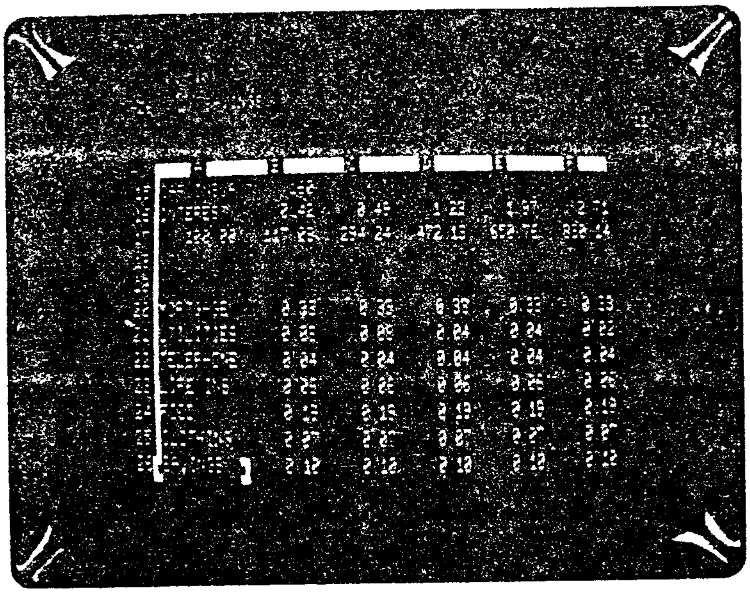
Type >**A11**Ⓔ⬍ to bring the titles back on the screen and highlight our initial SAVINGS figure at B11. Since the $160 expense for car insurance at B15 covers our premiums for six months, we should save one sixth of this amount each month, in addition to our usual savings deposit. Type **150+(B15/6)**Ⓔ (The parentheses in the formula tell the VisiCalc program to calculate that portion of the formula first. See the section entitled "The VALUE Command" in Part III for a full discussion of precedence.) Thanks to our earlier use of formulas, the VisiCalc program automatically propagates the adjusted savings figure across all twelve months, and also recalculates leisure and our new savings account balance. This recalculation has made the screen display somewhat messy, because the global format for numbers is still the standard format, or "general." We can clean up the display by typing **/GFI** The interest and savings account figures still show dollars and cents, because each of these entries has the "local format" /F$. Now, type >**N17**Ⓔ to reexamine our total discretionary income. It has increased to 6673.53. By saving about $27 more each month, we have reduced our leisure total back to 4730, but we have regained most of the interest we had lost before. The VisiCalc program can really help you budget in ways that you wouldn't have contemplated before!

Obtaining Monthly Expense Percentages

This lesson has given you a lot of practice in techniques for using commands such as REPLICATE effectively. Here's a challenge to test your mastery of the REPLICATE command: Give yourself monthly percentages of income for each of your expenses, from mortgage through savings. It's possible to accomplish this by typing just one formula and using the replicate command twice. Before trying, save your work by typing **/SS** and using ⬍ as necessary to bring the file name LESSON3B/VC:0 onto the edit line, and press **CLEAR** six times, then type **C/VC:0**Ⓔ (If you need them, here are some hints: Use the area of the sheet directly below your list of monthly expenses. Remember that you can replicate format specifications. If you want to get fancy, you can label each row of percentages with one more use of the REPLICATE command.)

If you've succeeded, congratulations. You can probably do anything you want with the VisiCalc program from now on. Whether you've succeeded or not, let's make sure that your budget matches the one in this lesson: Clear the sheet with **/CY** and reload the file you just saved by typing **/SL** then pressing ♦ until the file name LESSON3C/VC:0 appears, and then pressing Ⓔ Now type exactly the following:

```
>A20Ⓔ
/R CLEAR A5.A11:A20:
♦/F$+B5/B3Ⓔ
/R:B21.B26:RN
/R.B26:♦.M20:RRRRRRRRRRRRRR (Type R fourteen times)
>A26Ⓔ
```



The first replication in this sequence illustrates another use of the replicate command: You can copy a range of entries into another part of the same column (in fact, to any other area on the sheet) simply by specifying the destination starting coordinate for the target range. Check row 21, for utilities, to verify that the percentages may change from month to month.

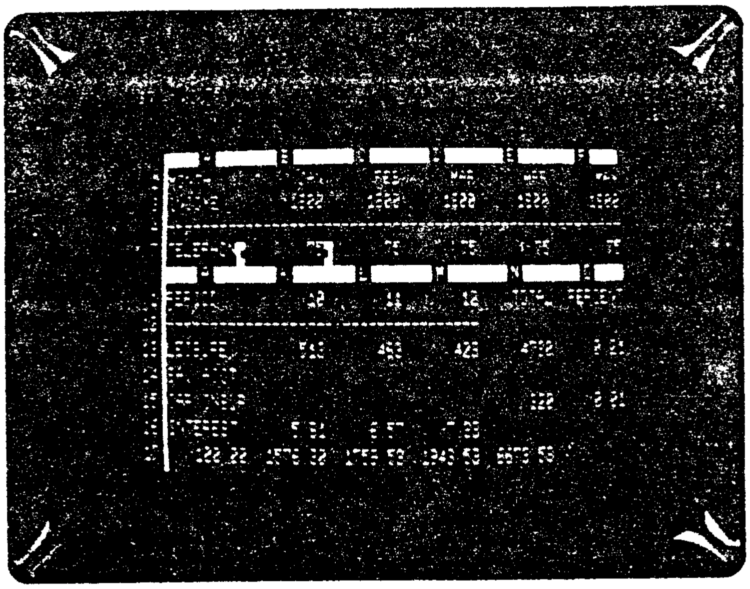
Synchronized Scrolling

As you can see, the area of the sheet which we have used extends beyond the screen window in both the horizontal and vertical directions. The window has begun scrolling downwards, and our month labels, income, and our first few expenses have disappeared from view. Let's split the screen so that we can see both the expense amounts and the percentages at the same time. Move the cursor up to A19, just above MORTGAGE. Now type **/WH** The screen splits horizontally, leaving just enough room for the expense percentages in the bottom window (which we'll attend to in a moment). Now type **>A2**Ⓔ↓↓ This should leave rows 2 (MONTH) through 5 (MORTGAGE) on display in the top window, with the cursor at A4. Next, type **/TB** to fix the horizontal and vertical borders in place. (Notice that the columns and rows forming the borders do not have to start from the edges of the sheet.) Finally, type **;>A26**Ⓔ to bring all of the expense percentages into view, and **/TV** to fix the labels MORTGAGE through SAVINGS in place.

Now press → until the bottom window begins moving to the right. The problem is that we can't easily tell which months these expense percentages refer to. The month labels, such as JAN, FEB and MAR, are visible only in the top window, which isn't scrolling. We'd really like the two windows to scroll together horizontally, but remain independent vertically so that we can view different areas of the sheet through the two windows, as we are now (with rows 2–5 in the top window and rows 20–26 in the bottom). Press /W again, and look at the prompt line. The possible keystrokes are:

H	To split the screen horizontally.
V	To split the screen vertically.
1	To return to one screen window.
S	To synchronize the two windows.
U	To unsynchronize the windows.

Press **S** and watch what happens in the **top** window. Notice that this window scrolls over so that portions of the same columns are visible through the top and bottom windows. Now press ↕ a few times: The two windows move together "in sync." Type ; to jump the cursor into the top window, and then type **>B7Ⓔ** Try changing the telephone expense to something else, say $100 per month, and watch the line of percentages opposite TELEPHONE in the bottom window. Now, let's use the two windows for a different purpose. Type **/WU** to unsynchronize the windows. Then move to the bottom window with ; and type **>A1Ⓔ/TB>O17Ⓔ** to bring your leisure total and percentage, your final savings account balance, and your combined discretionary income into view. Then press ; to jump back to the top window, landing at B7, and change the telephone expense back to 75, watching how this effects the leisure percentage in the bottom window. You can also experiment with the budget in other ways. At this point, you may wish to save the results of this lesson by typing, for example, **/SSMYBUDGET/VCⒺ**



The Order of Recalculation

So far, we've simply noticed that the VisiCalc program recalculates the values of all the formulas on the sheet, but we haven't looked closely at how this is done. There are some subtleties to the process of recalculation which can affect your results when you set up a complex problem with many interdependencies. We'll consider these issues here.

The VisiCalc program recalculates by starting at the upper left hand corner of the sheet and working its way downward and to the right until it reaches the lower right hand corner of the sheet. Each formula is evaluated only once, unless you ask for an extra recalculation by pressing !

The VisiCalc program will evaluate the formulas on the sheet in either of two possible orders: "down the columns" or "across the rows." Look again at the letter C in the upper right corner of the screen. This is the **recalculation order indicator,** and it can be either C, for columnwise recalculation, or R, for rowwise recalculation. When you load the VisiCalc program or clear the sheet, the program is set to recalculate "columns first:" It will evaluate first A1, then A2, A3, . . . , etc., then B1, B2, B3, . . . , etc., then C1, and so on. If you change the recalculation order to "rows first," the VisiCalc program will evaluate first A1, then B1, C1, . . . , etc., then A2, B2, C2, . . . , etc., then A3, and so on.

For many problems, the choice of row versus column recalculation has no effect on the results displayed on the screen. But there are cases where you must use the right recalculation order to obtain correct results, and it's important to recognize these cases when they arise. So let's consider an example. Clear the sheet with /CY and type the following:

```
1▶−A2Ⓔ
>A2Ⓔ
1+A1▶2*B1Ⓔ
>C5Ⓔ
+A1▶1+C5Ⓔ
>C6Ⓔ
−D5▶2*C6Ⓔ
>A1Ⓔ
```

As you type the formulas, think about how each entry depends on the other entries. As you can probably see, the matrix of entries starting at A1 must be recalculated in the order A1, A2, B1, B2 (since B1 depends on A2). The matrix of entries at C5, however, must be recalculated in the order C5, D5, C6, D6 (since C6 depends on D5). With the cursor at A1, type 2Ⓔ and watch what happens. A2 becomes 3, B1 becomes −3, and B2 becomes −6, as expected; but while C5 becomes 2 and D5 becomes 3, C6 remains at −2, and D6 at −4. The formula at D5 **was** recalculated, but this occurred **too late** to affect the recalculation of C6 and D6. Now press ! to trigger an extra recalculation: This time C6 becomes −3 and D6 becomes −6.

Now we'll change the order of recalculation from "columns first" to "rows first:" Press **/G** The prompt line again reads Global: C O R F. Press **O** Now the prompt line reads Reeval Order: R C. Press **R** Notice that the recalculation order indicator at the upper right corner of the screen changes from C to R. Now type **1**Ⓔ and watch the screen. This time, D5 becomes 2, C6 becomes −2, and D6 becomes −4; but while A2 becomes 2, B1 stays at −3, and B2 at −6. Our problem with B1 and B2 is, of course, similar to our earlier problem with C6 and D6.

The moral of this example is that you should lay out your calculations with either columnwise or rowwise recalculation in mind, but not both. If possible, you should arrange things so that the results will be correct with **either** columnwise or rowwise recalculation. Then, if you decide to calculate something new that requires a particular order of recalculation, you won't be constrained by other dependencies on the recalculation order. For example, the personal budget outlined in Lesson Three is independent of the order of recalculation. Now, suppose that you want to adjust your life insurance policy and premiums to provide a benefit of three times your total annual income. You could do this by switching to "row first" recalculation, so that the life insurance premium in month 1 could be based on the total income calculated in column N.

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

Forward and Circular References

Clear the sheet with **/CY** and type the following: **1⬗−⬗**Ⓔ The entry contents line reads B1 (V) −C1, and the value display at B1 is 0, as expected. Now type **⬗1+⬖⬖**Ⓔ The entry contents line reads C1 (V) 1+A1, and the value at C1 is 2, while the value at B1 has been updated to −2. Next, press **>B2**Ⓔ**−C1**Ⓔ We now have the same formula, −C1, at both B1 and B2, and both positions display the value −2. Is there any difference between these two formulas? Indeed there is. One of these formulas will recalculate and display the correct value only if the order of recalculation is "row first." The other will never display the correct value after an automatic recalculation! To see this, type **>A1**Ⓔ**2**Ⓔ and watch the screen. A1 becomes 2 and C1 becomes 3, but both B1 and B2 remain at −2. Press **!** and both B1 and B2 will be updated to −3. Now, type **/GOR** and note that the recalculation order indicator changes from C to R. Then type **3**Ⓔ and watch the screen. A1 becomes 3, C1 becomes 4, and now B2 becomes −4, but B1 remains at −3. You'll have to press **!** again before B1 will change to −4. And if you change A1 again, B1 will display a value based on the previous contents of C1 and A1. The formula at B1 is an example of a **forward reference**: It contains a reference to an entry which will be recalculated after B1 is recalculated, regardless of whether rowwise or columnwise recalculation is used.

In pathological cases, forward references may refer to other forward references, so that correct results can be obtained only with several recalculations. For example, press **▸/IC–▸Ⓔ** At the moment, A1 is 3, the new B1 is 4, C1 is –4, and D1 is 4. Now press **◂1Ⓔ** A1 becomes 1, D1 becomes 2, but B1 and C1 are unchanged. Press **!** Now C1 is –2, but B1 is still 4. Only after you press **!** again will B1 be updated to 2.

An effect even more startling is caused by a **circular reference**. The value of such a formula cannot be settled with any number of recalculations! Clear the sheet with **/CY** and type the following: **1+▸Ⓔ** The entry contents line reads A1 (V) 1+B1, and the value in the cursor is 1, as expected. Now type **▸1+◂Ⓔ** and watch carefully. What happened? The numbers at A1 and B1 actually changed **twice**. When the formula 1+A1 was completed at B1, it was evaluated, yielding 1+1 or 2 at B1. Then, since the value of B1 had been changed, an automatic recalculation occurred. A1, or 1+B1, became 1+2 or 3, and B1, or 1+A1, became 1+3 or 4. Now press **!** A1 increases to 5, and B1 becomes 6. These values will change every time you press **!**

The foregoing examples have been somewhat artificial: You probably recognized the forward and circular references as soon as you typed them. If you are planning your work carefully, you shouldn't write such a formula unless you do it purposely, as, for example, in preparing a debugging tool. /SS :P, saving formulas to the printer, is helpful in debugging circular references. But some forward or circular references are not so obvious. 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.)

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

Summary

Once again, we have covered a great deal of ground in this lesson. We concentrated on a variety of techniques for using the VisiCalc program's flexible REPLICATE command as effectively as possible. We also discussed a number of new VisiCalc features, including the formatting of individual numbers and labels, and the relationship between global and local formats; the order in which the VisiCalc program calculates and recalculates; built-in functions such as @SUM, and the special cases @NA and @ERROR; the insert, delete and move commands, which let you "stretch" the sheet and manipulate entire rows or columns; and synchronized scrolling of the two screen windows. By this time we have covered most of the important concepts and features of the VisiCalc program, and you should be able to use the VisiCalc program to considerable advantage. In Lesson Four we will cover a number of other features and fine points of the VisiCalc program, including scientific notation for numbers and the transcendental functions such as @EXP, @LN, @SIN, and @COS; more about the formatting of numbers, and the move command. You can go on to Lesson Four as soon as you are ready. But now is an excellent time to try out what you have learned on a problem of your own. In this way, you will consolidate your knowledge of the VisiCalc commands and features, and develop a measure of intuition about how they can be used. This will enable you to approach new problems and solve them even more rapidly with the VisiCalc program.

Postscript: The Print Command

Sometimes it's convenient to have a "hard copy" of the contents of the electronic sheet on a real sheet of paper. The VisiCalc Print command lets you print all or any part of the electronic sheet on a printer. If you have a printer connected to your computer, you can try it out by printing a copy of the personal budget we've just created. The instructions for printing files you have made with the VisiCalc program are in Part III of this manual, in the section entitled "The PRINT Command".

Lesson Four

Lesson Four

Lessons One, Two and Three have covered most of the essential features of the VisiCalc program. In this lesson, we'll be primarily concerned with features which extend the VisiCalc program's usefulness in applications where complex or lengthy formulas, numbers with very large or small magnitudes, or arithmetic operations other than simple addition, subtraction, multiplication and division are required. If you intend to use the VisiCalc program for business or financial applications, you can skim much of this material, concentrating on topics of interest such as the @NPV (Net Present Value) and @LOOKUP functions and the ability to create bar graphs using the /F* formatting option. If you have scientific or engineering applications in mind, you'll find this lesson particularly relevant to your needs.

More on Numbers and Formats

In Lessons Two and Three we illustrated some of the ways you can control the display of numbers on the screen with formatting commands such as /GFI and /F$. In this lesson we'll examine the formatting options more closely.

Load the VisiCalc program into your computer (as described in Part I in the section entitled "Loading the VisiCalc Program") or, if you already have the program running, clear the sheet with **/CY** When you clear the sheet, the "global format" is set to general. This is the effect you obtain when you type the command **/GFG** Each individual entry is set to **default** to the global format, just as it would if you had typed **/FD** with the cursor at that entry. Now type the following:

123.456Ⓔ
/R:◆.◆◆:

We now have the same number, 123.456, on display in three entry positions, A1, B1 and C1. Since we have not yet set any explicit formats, all three entries default to the global format. The global format, **general**, displays numbers in whatever form will show the value of the entry with the greatest precision. As you have seen, however, this may not be the most readable way to display a column of numbers.

Now press the following keys: **/FI▶/F$◀** We have set the "local format" of entries A1 and B1 to be **integer** and **dollars and cents**, respectively. The local format setting at the entry where the cursor lies, A1, is visible on the entry contents line, which reads A1 /FI (V) 123.456. Each entry is displayed in rounded form. At A1, for example, 123.456 is rounded to 123, since .456 is less than .5; but at B1, 123.456 is rounded to 123.46, since the last digit .006 is greater than .005. At C1, we still have 123.456.

Now type **/GFI** to change the global format from **general** to **integer**. Entries A1 and B1 are unaffected, because they have explicit local formats; but C1 now displays 123. With the cursor still at A1, change the local format there by typing **/FG** The number at A1 now appears as 123.456, and the entry contents line reads A1 /FG (V) 123.456. The local format overrides the global format integer. Next, press ▶ to move to B1, and "erase" the local format there by typing **/FD** This causes the display format of B1 to default to the global format, which is currently **integer**; so the entry at B1 appears as 123. The entry contents line reads B1 (V) 123.456; the explicit format setting /F$ has disappeared. Finally, type **/GFG** to set the global format back to general. Now all three entries display the number as 123.456. Position A1 has a local format which overrides the global setting, but the local format is also general. Positions B1 and C1 have no explicit format, so they default to the general format.

The way in which numbers are displayed in the general format depends on the column width. Type /GC7Ⓔ Now the number appears as 123.46 at all three entry positions. The VisiCalc program always allows for one blank at the left end of the entry position, and then displays as many significant digits as it can. To compare the flexibility of the general format to an explicit local format, try the following: **>B1**Ⓔ and then type **/F$** to set dollars and cents format at position B1. Then type /GC6Ⓔ Positions A1 and C1 now display the number as 123.5, but B1 now shows >>>>> (an effect which we saw before in Lesson Two). The VisiCalc program is telling us that it cannot display a number as large as 123.456 with two decimal places in a column six characters wide. If you type **12.34**Ⓔ the VisiCalc program will be able to display this number at B1.

Scientific Notation

Type **/CY** to clear the sheet. At position A1, type **99999999** (that's eight times) followed by ► This is the largest number that we can display, with a leading blank, in a nine-character column. At B1, type **1+◄Ⓔ** followed by ► The calculated result, 1+99999999 or 100000000, is too large to display in ordinary form at B1, so the VisiCalc program has switched to **scientific notation**: The number appears as 1E8, meaning "1 times 10 to the 8th power" or 1 followed by 8 zeros. Scientific notation can also be used to display very small numbers. At C1, type **.000000001** (that's eight zeros) followed by ► The number appears as 1.E-9, meaning "1 times 10 to the −9th power," or "1 with the decimal point moved left 9 places." (Take a moment to convince yourself that this is correct.) Now at D1 type **−◄Ⓔ** The result is displayed as −1.E-9.

When a number is displayed in the general format, the VisiCalc program will automatically shift between conventional and scientific notation as required to display the calculated value with the greatest precision. Type **/GC12Ⓔ** Notice that all of the values revert to conventional notation in the widened columns: at D1, for example, the number appears as −.000000001. Next, type **/GC7Ⓔ** and **>A1Ⓔ** The number 99999999 at A1 has been rounded up and displayed as 10.0E7, or "9.9999999 (rounded to 10.0) times 10 to the 7th power." Finally, type **/GC5Ⓔ** and look at the screen. In order to fit the numbers in these narrow columns, the VisiCalc program has eliminated decimal points, displaying A1 as 10E7 and C1 as 1E-9; but D1 shows >>>> since there's not enough room in a five-character column to display −1E-9 (with a leading blank).

More on Value References

Clear the sheet with **/CY** and type the following:

1►2►+A1/B1Ⓔ

The entry contents line reads C1 (V) +A1/B1, and the value displayed is .5. We know that if the number at either A1 or B1 were changed, the formula at C1 would be recalculated. Now press ► to move to D1. Here we'll try something different. Press **+A1/B1#** watching the edit line as you do so. As soon as you press the # after the coordinate B1, the reference to B1 is replaced by its current value. The edit line now reads +A1/2. Press Ⓔ The entry contents line reads D1 (V) +A1/2, and the value displayed is again .5. The difference is that the current value of B1 is "fixed" into the formula at D1, whereas the value of B1 is changeable at C1. To verify this, type **◄◄4Ⓔ** and notice that C1 changes to .25, but D1 does not.

The effect of # after a value reference on the edit line is similar to the effect of typing ! after a formula, as described in Lesson One. The difference is that the ! key evaluates the entire formula on the edit line, replacing it with a single number, while the # key fixes the value of a single coordinate, so that the rest of the formula can contain changeable elements. If the # is not preceded by a coordinate such as B1, it is replaced by the current value of the entry where the cursor lies (i.e., the entry you are changing). You can use this feature to take a look at the precise value of a formatted entry on the edit line. For example, set the global format with **/GF$** and then type **6** Ⓔ The value at C1 is now displayed as 0.17. Press ▸ to move to C1, then press # The # immediately places the current value of C1 (1/6) to maximum precision (.166666666666) on the edit line. It freezes numbers at their current value.

A word on precision is in order here. The VisiCalc program maintains numbers internally in decimal form. Certain fractions (such as 1/6) cannot be expressed exactly with any fixed number of significant digits. The VisiCalc program uses decimal based arithmetic so that it can maintain accuracy in calculations involving dollars and cents. To accommodate large financial figures as well as high-precision engineering or scientific quantities, the VisiCalc program guarantees precision to eleven digits (and sometimes twelve digits) at base 10. The last 6 on the edit line at the moment is a "guard digit," which allows the VisiCalc program to determine which way to round the eleventh digit when a calculation is completed. After examining the number, you can press **CLEAR** a few times to "abort" the VALUE entry you have started on the edit line.

More on Formulas

In the examples from previous lessons, we have used only simple formulas (or **expressions**) whose meanings have been clear. As you begin to write more complex expressions involving several arithmetic operations, the way in which such expressions should be evaluated may not be so obvious. For example, to evaluate the expression 9+6/3, should we first add 9 to 6 giving 15 and then divide by 3 to obtain 5; or should we first divide 6 by 3 giving 2, and then add 9 to obtain 11? Try it: First type **/CY** to clear the screen and then type **9+6/3**Ⓔ The answer displayed at position A1 is 5. Evidently the VisiCalc program chose the first option. In this way, the VisiCalc program is similar to many keystroke calculators in that it always evaluates expressions strictly **from left to right.**

You can change the order of evaluation of arithmetic operations in an expression by using parentheses. For example, press ↓ to move to A2 and then type **9+(6/3)**Ⓔ The answer highlighted by the cursor is 11. Parentheses may be nested to any depth. Type

↓−(↑↑+((A2−1)/A1))Ⓔ

The answer displayed at A3 should be −7.

More generally, an expression consists of a series of **operands** separated by arithmetic **operators**. Each operand can be one of the following:

1. A number, optionally with a decimal point and/or an E exponent.
2. A value reference, obtained either with cursor movements or by typing the coordinate.
3. A function reference, with zero or more arguments in parentheses.
4. An expression surrounded by parentheses. (Such a subexpression is evaluated first.)
5. Any of the above, preceded by a − or + sign.

Each operator can be one of the following:

+	For **addition.**
−	For **subtraction.**
*	For **multiplication.**
/	For **division.**
∧	For **exponentiation.**

On your keyboard, you generate the exponentiation symbol (∧) by holding down the **SHIFT** key and pressing the @ key. The exponentiation operator ∧ lets you calculate "powers." For example, type **↓2∧3**Ⓔ to calculate "2 to the 3rd power," or 8. Try another example: Type **↓2∧.5**Ⓔ to calculate "2 to the power 1/2," or the square root of 2. The result displayed at A5 should be 1.414214. To find the cube root of 5, type **↓5∧(1/3)**Ⓔ The VisiCalc program will calculate and display the value 1.709976.

More on Built-in Functions

We first encountered built-in functions in Lesson Three, where we used the @SUM function to find yearly totals for our income and expenses and to calculate our available leisure money as +B2−@SUM(B4.B11), or income minus the sum of expenses. As mentioned above, a function reference can appear in an expression wherever a number or coordinate could appear. Moreover, as we shall see shortly, an expression may also occur as an argument in a function reference. We also saw that one type of function argument is a range of entries, such as B4 . . . B11 in the LEISURE example. You need only type one period; the VisiCalc program will fill in the other two.

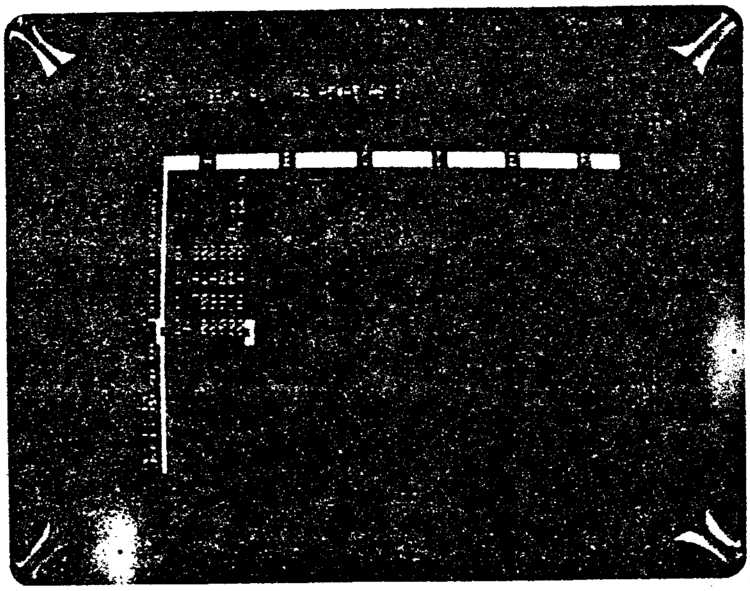
In general, a built-in function name is followed by a parenthesized **list** of **arguments** separated by commas. Each argument can be:

1. An **expression**, i.e., a series of numbers, value references, and/or function references separated by arithmetic operators and/or parentheses; or
2. A **range** of entries, i.e., a series of entries that are next to each other in a row or column, such as B2, B3 and B4, or B2, C2, D2, and E2. A range is specified by typing (or obtaining with cursor movements) the first and last entry separated by an ellipsis (. . .). For example, the ranges just mentioned would be specified as B2 . . . B4 and B2 . . . E2, respectively.

The exact number and type of argument(s) required varies from function to function. For example, the @NA and @ERROR functions which we saw in Lesson Three required no arguments. Some functions require exactly one or two arguments, while others, such as @SUM, take a variable number of arguments.

Let's try an example of the @SUM function using the general form for a list of arguments. Use the ⬅ key to move to position A7, and type:

```
@SUM(A1.A4,A5*A5,A6∧3)Ⓔ
```



The result should be 5+11−7+8+2+5 or 24. Let's experiment further with some other built-in functions:

@MIN and @MAX. These functions accept a list of arguments just like the @SUM function. The result is the minimum and maximum value in the list, respectively. Remember that the minimum value will be the negative number (if any) with the greatest absolute magnitude. To try out these functions, type the following:

```
⬅@MIN(A1.A7)Ⓔ
⬅@MAX(A7,@SUM(A1,A2,A4.A6))Ⓔ
```

The results should be −7 for @MIN at A8, and 27.12419 for @MAX at A9.

@COUNT and @AVERAGE. These functions also accept a list of arguments, which may be expressions or entry ranges. @COUNT determines the number of nonblank entries occurring in the range or ranges of the argument(s). (Note that arguments which are expressions rather than ranges always count as 1. This can be puzzling if the "expression" is a single coordinate such as B1: It will add 1 to the count **even if B1 is blank**. To avoid this, write B1 . . . B1.) @AVERAGE finds the arithmetic mean of the entries making up the argument(s); it is equivalent to @SUM(**arguments**)/ @COUNT(**arguments**). To try out these functions, clear the sheet with **/CY** and type the following:

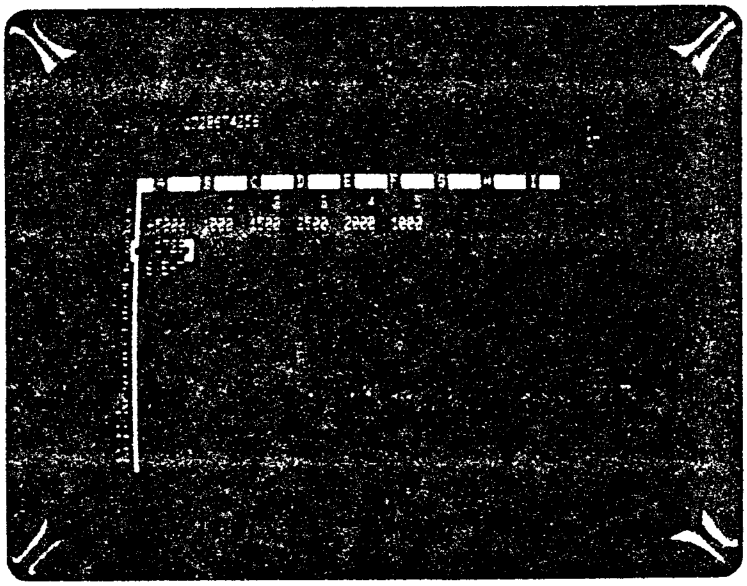
1↓↓3↓4↓↓6↓↓8↓/−−↓

The cursor should now be at A10. Type **@COUNT(A1.A8)Ⓔ** The count of nonblank entries displayed at A10 should be 5. Now type **↓@ AVERAGE (A1.A8)Ⓔ** The average of 1, 3, 4, 6 and 8 should be 4.4 at A11. To check the equivalence mentioned above, type **↓@SUM(A1.A8)/A10Ⓔ** The result should again be 4.4. Finally, let's change one of the currently blank entries in the argument range to a number: Type **>A5Ⓔ5Ⓔ** The @COUNT at A10 should change to 6, and the @AVERAGE at A11 and A12 should increase to 4.5.

The @NPV (Net Present Value) Function. This function accepts two arguments. The first is a single expression specifying a discount rate (DR), such as .15 for 15% per period. The second argument is a range of entries; the first entry is the cash flow at the end of the first period, the second entry is the cash flow at the end of the second period, etc. The result of the function is the net present value of the cash flows in the range, discounted at the rate specified by the first argument. If we represent the function reference by @NPV(dr,entry**1** . . . entry**n**) and let DR=1+dr, the result of the function is (entry1/DR) + (entry2/(DR∧2)) + (entry3/(DR∧3)) + . . . + (entryN/ (DR∧N)). To illustrate the use of this function, type **/CY/GC6Ⓔ** to clear the sheet and make narrower columns. Suppose that we have a project which requires an investment or cash expenditure of $5000 up front, and which is expected to generate cash over a period of five years. Type the following to lay out the cash flows:

→1→1+←Ⓔ
/R:D1.F1:R
>A2Ⓔ
−5000→1000→1500→2500→2000→1000Ⓔ
>A3Ⓔ
.15↓

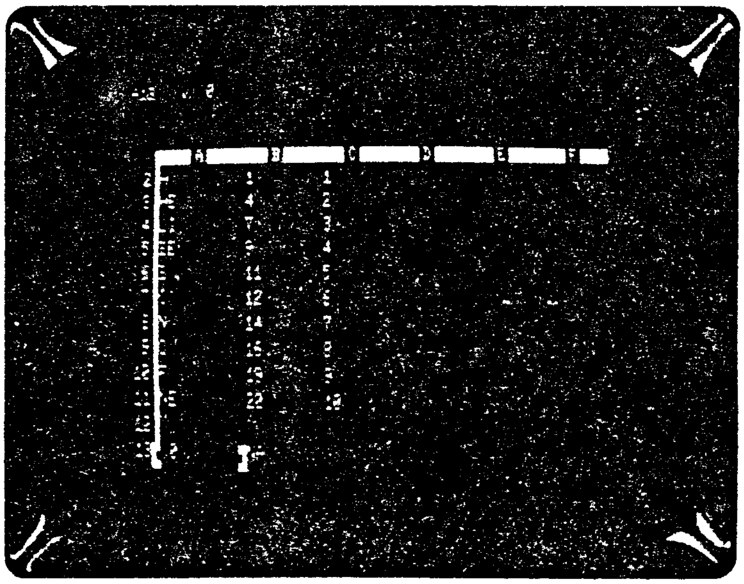
At position A4, type **+A2+@NPV(A3.B2.F2)**Ⓔ This causes the $1000 cash flow in the first year to be discounted by 15%, the $1500 cash flow in the second year to be discounted twice, etc. The result at A4 should be 288.3. Now press ← and change the discount rate to 10% by typing **.1**Ⓔ The net present value at the lower discount rate (which makes the future cash flows worth more today) is 1014. You can find the internal rate of return of this project by trial and error: Try higher and lower discount rates at A3 until the net present value of the cash flows is near zero. When A3 is .17, A4 will be 34.81. (If you are persistent, you can verify that the net present value is .00000061 when the discount rate is .1728674256.)



The @LOOKUP Function. This function accepts two arguments. The first or "search" argument, an expression, is looked up in the table specified by the second argument, an entry range, and matched against one of the entries in this range. A table of function result values must be present in the column or row immediately to the right of or below the column or row range specified in the @LOOKUP function. The function result is taken from the entry corresponding to the matched entry in the table of result values. The values in the function argument range are ordinarily in ascending order. The search argument is compared against succeeding entries in the second argument range, until an entry greater than the search argument is found. The search argument is "matched" against the entry just before this one (i.e., one which is still less than or equal to the search argument), and the function result is selected from the corresponding entry in the table of result values. If the **first** entry in the argument range is greater than the search argument (i.e., if the search argument cannot be matched against any entry in the range), the result of the function is NA or Not Available. To illustrate the use of this function, we'll list the first ten entries in the periodic table of the chemical elements, with their atomic weights and atomic numbers. Clear the sheet with /CY/GFL and type the following:

ELEMENT→WEIGHT→NUMBERⒺ
>A2Ⓔ
H↓HE↓LI↓BE↓B↓C↓N↓O↓F↓NEⒺ
>B2Ⓔ
1↓4↓7↓9↓11↓12↓14↓16↓19↓20Ⓔ
>C2Ⓔ
1↓1+↑Ⓔ/R:C4.C11:R
>A13Ⓔ
10.9→@LOOKUP(←,B2.B11)Ⓔ

The result of the @LOOKUP function at B13 should be 4. Given an experimental atomic weight of 10.9, the @LOOKUP function compared this value against successive values in column B, stopping at the value 11 at B6 which was greater than 10.9. Thus, the atomic weight of 9 at B5 is the matching value, and the corresponding entry, the atomic number in column C, is 4. If you change the value to be looked up with **←12.1**Ⓔ the function result will become 6. If you then type an "atomic weight" of **0**Ⓔ the result of the @LOOKUP function will be NA.



The @ABS and @INT Functions. Both of these functions accept a single argument, an expression. @ABS finds the "absolute value" of its argument: For example, @ABS(1)=1, @ABS(−1)=1, and @ABS(0)=0. @INT finds the "integer portion" of its argument, without any rounding. You can think of the @INT function as setting every digit to the right of the decimal point to zero. For example, type **/CY/FI1.7▶@INT(◀)**Ⓔ At A1, the value 1.7 is rounded up to 2 by the local format integer; but at B1, @INT finds the integer portion of 1.7, i.e., 1.0 or 1.

Transcendental Functions and Graphing

So far, we've covered all of the built-in functions except for the transcendentals such as @EXP, @LN and @SIN, and all of the formatting options except for the "graph" format /F*. We'll illustrate these two features together with a more complete example. Our goal is to produce graphs of the transcendental functions. We'll have to do this within the limits of the /F* formatting option, which is really designed to draw simple bar graphs in a column alongside other columns of numbers.

Clear the sheet with **/CY** and type **1▶3▶6▶20**Ⓔ then **>A1**Ⓔ The "star" format /F* simply displays the number of asterisks equal to the integer portion of the value of the entry where the format is set. Type **/F*** at A1: In place of the (right-justified) number 1, a single (left-justified) asterisk appears, after the usual leading blank. Now press **▶/F*** The value 3 is replaced by three asterisks. Continue with **▶/F*▶/F*** Position C1 shows six asterisks, while position D1 shows eight (the maximum for a nine-character column). Type **/GC12**Ⓔ and notice that position D1 now displays 11 asterisks.

Now clear the sheet again with **/CY** To graph a function, we must first supply a series of argument values for the function and calculate the function result for each argument value. For simplicity's sake, we'll begin with a linear function, e.g., **f(x)=2.5 * x**. Type the following:

>A20Ⓔ
.1▶.1Ⓔ
>A1Ⓔ
+A20◀+▲+B20Ⓔ
/R:◀.A17:RN

On row 20, we have defined a "start" value and "step" value for our list of function arguments. Then, using formulas and the REPLICATE command, we calculated the argument values, from .1 to 1.7 in positions A1 to A17. Next, we'll calculate the corresponding function results for our linear function. Type the following:

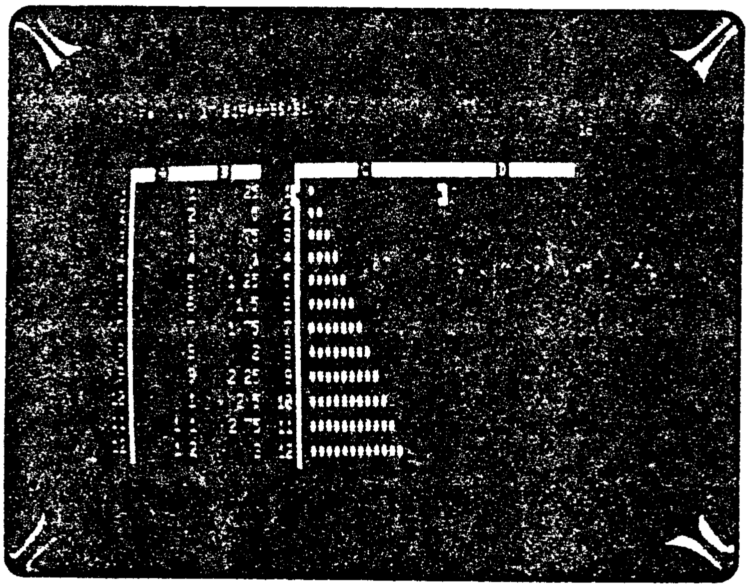
```
>B1Ⓔ
2.5*A1Ⓔ
/R:◆.B17:R
```

The function results are .25, .5, .75, etc., up to 4.25. To see how this looks in the star format, we'll set the format specification and replicate again: Type **/F*/R:◆.B17:R** How does the "graph" look? It's probably not the kind of graph you had in mind. The problem is that the function results do not fall conveniently in the range 1, 2, 3, etc., which would yield one, two, or three asterisks. Let's go back to a numeric display with **/FD/R:◆.B17:R** How can we create a better star format display?

First, we'll set up a wider column in which the asterisks may appear, so that we can represent a wider range of function results with the best possible resolution. This will allow us to display a "bar" of up to 7 asterisks. Then we'll "scale" the function results, from .25 to 4.25, into the range 0 to 17. Type the following:

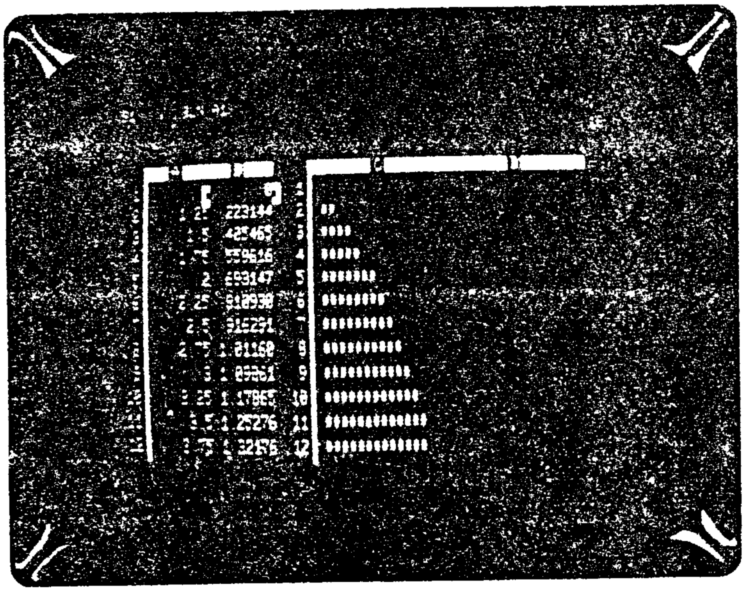
```
>B18Ⓔ
/--◆@MAX(B1.B17)Ⓔ
/GC8Ⓔ
>C1Ⓔ
/WV;/GC18Ⓔ
17/B19*@ABS(B1)Ⓔ
```

At B19, we've used the @MAX function to find the upper limit on the range of function results. (For simplicity's sake, we'll work with the absolute magnitudes of the function results, so the lower limit will be zero. You may wish to work out an approach to scaling which uses both @MAX and @MIN.) We've also set up a wide column in the right hand screen window for the asterisks. The formula at C1 is used to multiply each function result, such as B1, by the factor 17 (the maximum number of asterisks) divided by B19 (the maximum function result). The value of this formula will lie in the range from 0 to 17/B19*B19, or 17. Now type /F* to set the display format at C1, and type /R:←.C17:NR and watch the screen. Now we have a reasonable approximation to a straight line. Moreover, if we've done our job properly, we should be able to graph any set of function results in column B, not just the linear function **f(x)=2.5 * x**. (We will want to adjust the argument range so that we can graph an interesting portion of the function result range.) Your screen should resemble the photo below.



Now type the following:

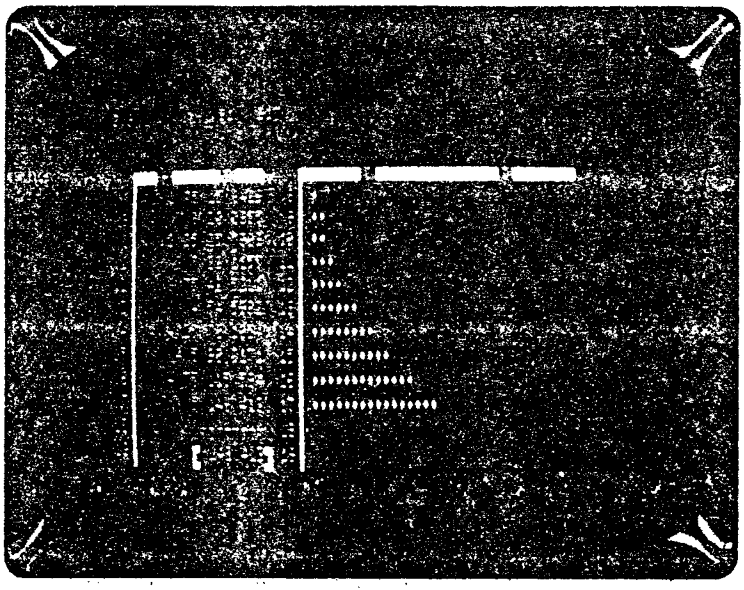
```
;>A20Ⓔ
1.25Ⓔ
>B1Ⓔ
@LN(A1)Ⓔ
/R:B2.B17:R
```



Notice that the value of @MAX(B1 . . . B17) at B19 changes to 1.60944. This value affects the formulas in column C so that the results still come out in the range 0 to 17. If the graph of the natural log function doesn't look completely familiar, tilt your head sideways and imagine the X axis on the column and the Y axis on the row.

Now, let's try the exponential function. Type:

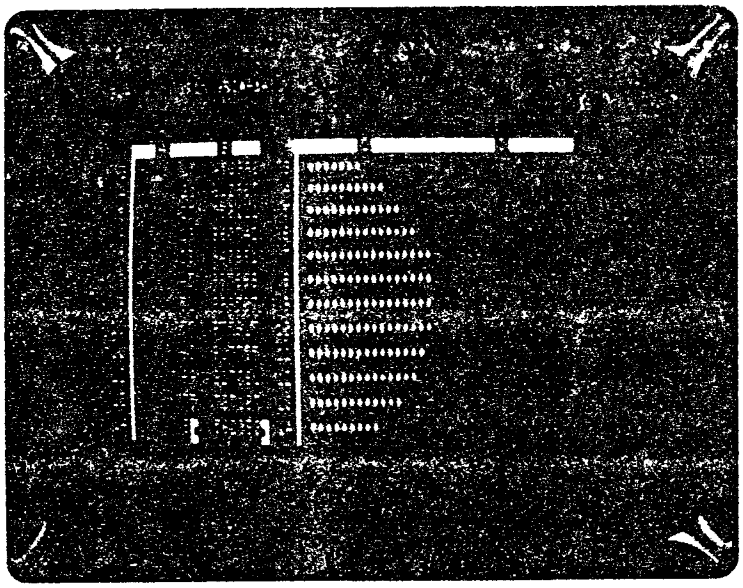
```
>B1Ⓔ
/WS
@EXP(A1)Ⓔ
/R:B2.B17:R
>B19Ⓔ
```



The maximum function result value is now 148.413, and each result is scaled into the range 0 to 17 in column C. Finally, we'll graph the sine function. Since the trigonometric calculations are done in radians, we'll use @SIN(@PI*A1) and a different argument range so that we can obtain a full sine curve in column C. Type the following:

```
>A20Ⓔ
.03▶.06Ⓔ
>B1Ⓔ
@SIN(@PI*A1)Ⓔ
/R:B2.B17:R
>B14Ⓔ
```

If all has gone well, you'll have a reasonable graph of the sine function in column C, looking like the photo below.



At this point, you may wish to experiment with different argument values and functions. When you're finished, you can go on with the next section.

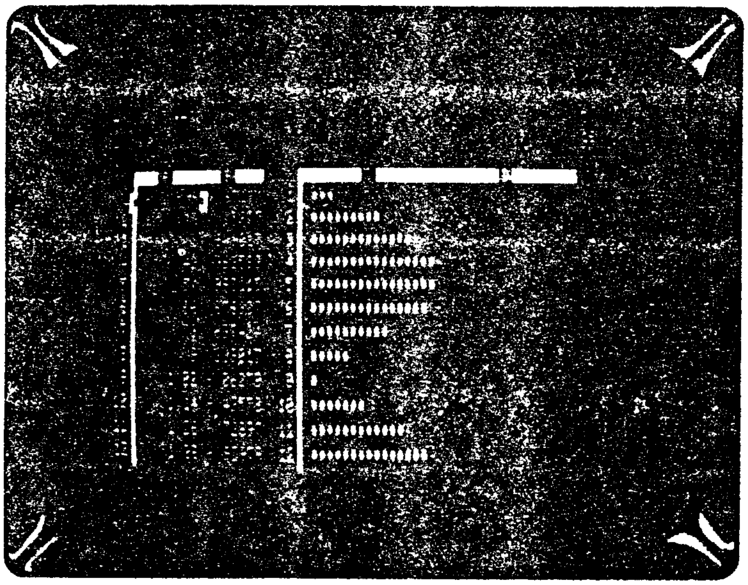
Manual and Automatic Recalculation

If you've just worked through the example above at your keyboard, graphing the transcendental functions, you've probably noticed some significant delays as the VisiCalc program repeatedly recalculated the results of functions such as @EXP, @LN and @SIN. Because the function results are calculated to nearly eleven significant digits, each one takes a fraction of a second to evaluate, and a sheet full of function references can take several seconds to recalculate. This problem gets worse, of course, as the amount of information on the electronic sheet increases.

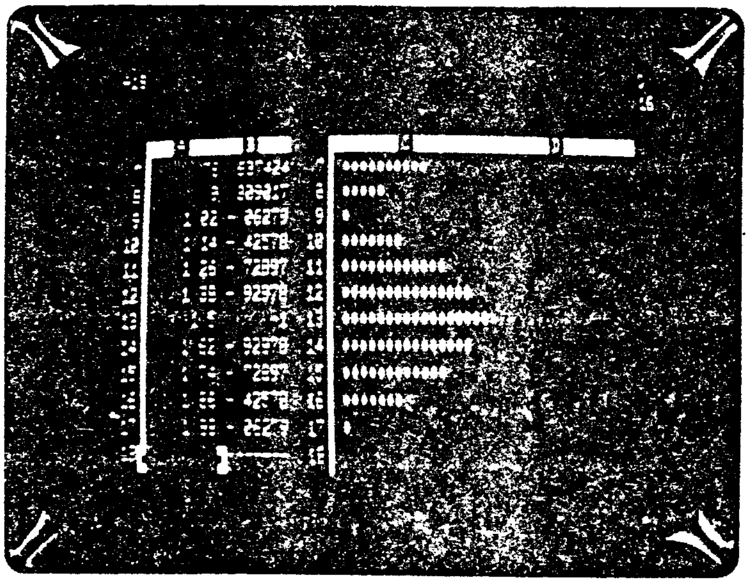
In many cases, you don't actually need to have all of the values recalculated every time you change an entry. It would be convenient if you could change several entries and then trigger a recalculation when you're ready to look at the results. The VisiCalc program lets you do this with the /GR command.

If you still have the graph of the sine function from the previous section on your screen, we'll try changing the argument range start and step sizes. (If you no longer have this graph on the screen, just read along.) Type >**A20**Ⓔ to move the cursor to the starting value, which is currently .03. When you load the VisiCalc program or when you clear the sheet, you are in **automatic recalculation** mode: As soon as you change the value at A20, the VisiCalc program will automatically recalculate the values of all the formulas on the sheet. Before doing this, however, type /**G** The prompt line reads Global: C O R F. Press **R** Now the prompt line reads Recalc: A M. The possible keystrokes are M, to switch to manual recalculation mode, and A, to return to automatic mode. Press **M** You have "turned off" automatic recalculation.

Now you can change the values at A20 and B20 without waiting for a lengthy intervening recalculation. Type **.06→.12**Ⓔ Now we're ready for a recalculation—but how do we make it happen? Press the exclamation key ! As you've probably noticed before, an exclamation point appears in the upper right corner of the screen while the recalculation takes place. Whether you're in manual or automatic mode, pressing ! triggers a recalculation of all formulas on the electronic sheet (unless ! is pressed while you're entering a LABEL or VALUE on the edit line). Notice how the graph changes to display the positive-going portion and the reflection of the negative-going portion of the sine curve. Type >A1Ⓔ



Now, type >B20Ⓔ.06 →.54 Ⓔ then return to automatic mode by typing /GRA. The first thing that happens as you return to automatic mode is, of course, a recalculation to update all the figures on the screen.



Summary

This lesson has introduced you to the full power of the VisiCalc calculation capability. Besides simple addition, subtraction, multiplication and division, the VisiCalc program provides exponentiation, transcendental functions, and scientific notation for numbers. You can use functions such as @SUM, @MIN and @MAX to manipulate entire rows, columns or other ranges of numbers at once. Functions like @COUNT, @AVERAGE, @NPV and @LOOKUP allow you to quickly handle common problems such as test score averaging, evaluating the terms of a loan, or looking up figures in the income tax tables. You can control the format of calculated results in a variety of ways—even creating simple graphs with the /F* format. With practice, you'll be able to use the features described in this lesson in combination with the VisiCalc screen and window control and formula replication capabilities to solve complex problems quickly and easily.

Lesson Four concludes Part II, the Tutorial for the VisiCalc program. Look up commands, as you use them, in Part III, the VisiCalc Command Reference. You will discover still more fine points of the VisiCalc program not discussed here, and many straightforward examples which illustrate how to use each command. As you use the VisiCalc program, you will discover countless techniques and application ideas that may be useful in your work.

PART III. VISICALC COMMAND REFERENCE

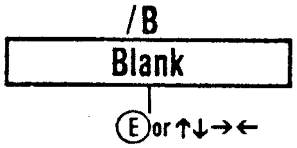
Introduction

Part III gives detailed summaries of all the VisiCalc commands as well as information on topics such as the VisiCalc screen, your keyboard, and file names.

The commands and topics are alphabetically organized. Command options are arranged in the order they appear on the prompt line of the VisiCalc screen. If you are not using the VisiCalc Tutorial, we recommend that you turn first to the section titled The VisiCalc Screen for general information about the VisiCalc program.

To help you find specific information, each command reference begins with a chart displaying all of that command's options. The charts for the individual commands are taken from the Command Structure Chart on the next page. This chart makes a very handy "roadmap" to maximum productivity with the VisiCalc program.

The BLANK Command

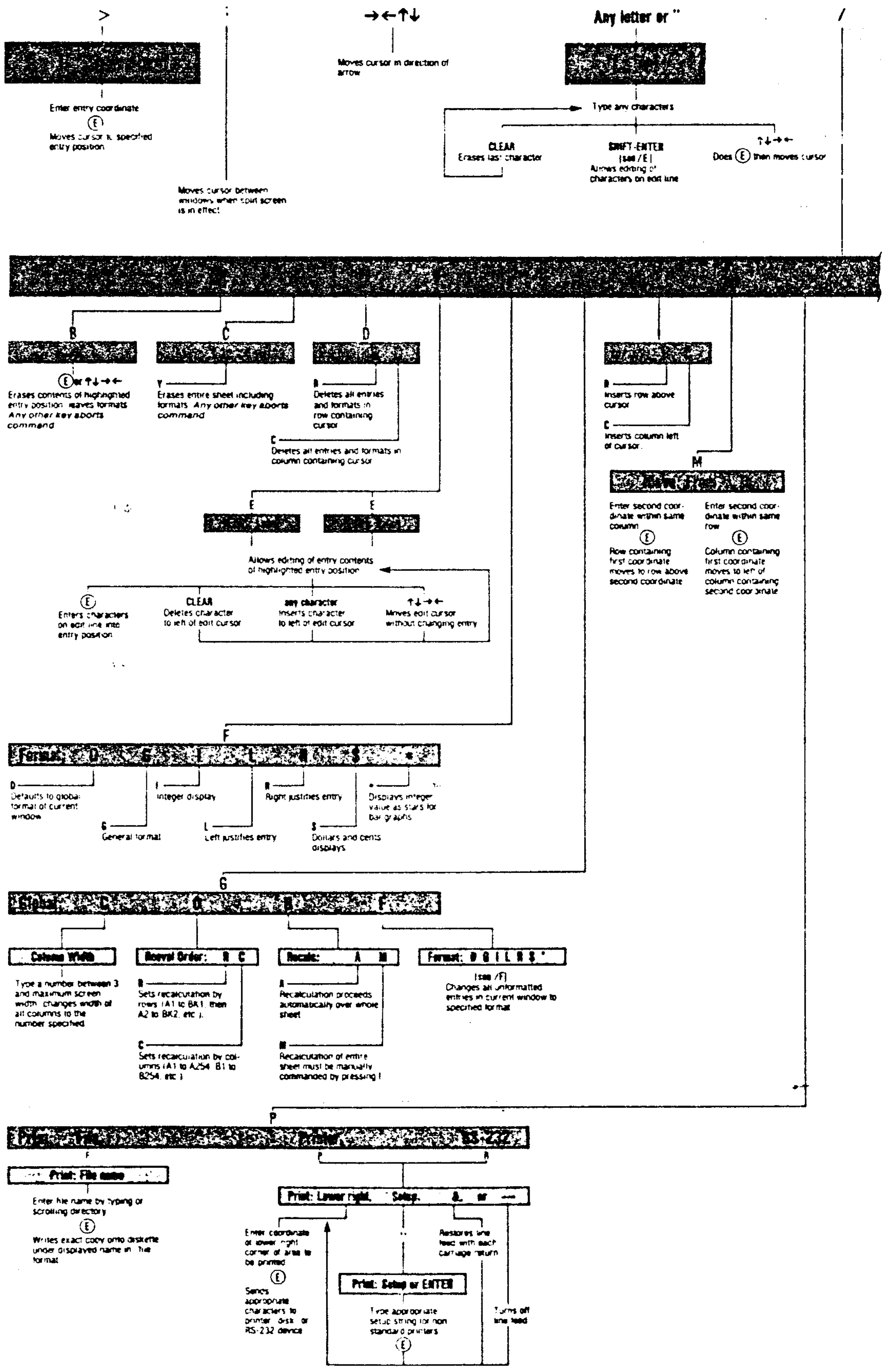


Erases contents of highlighted entry position, leaves formats.
Any other key aborts command.

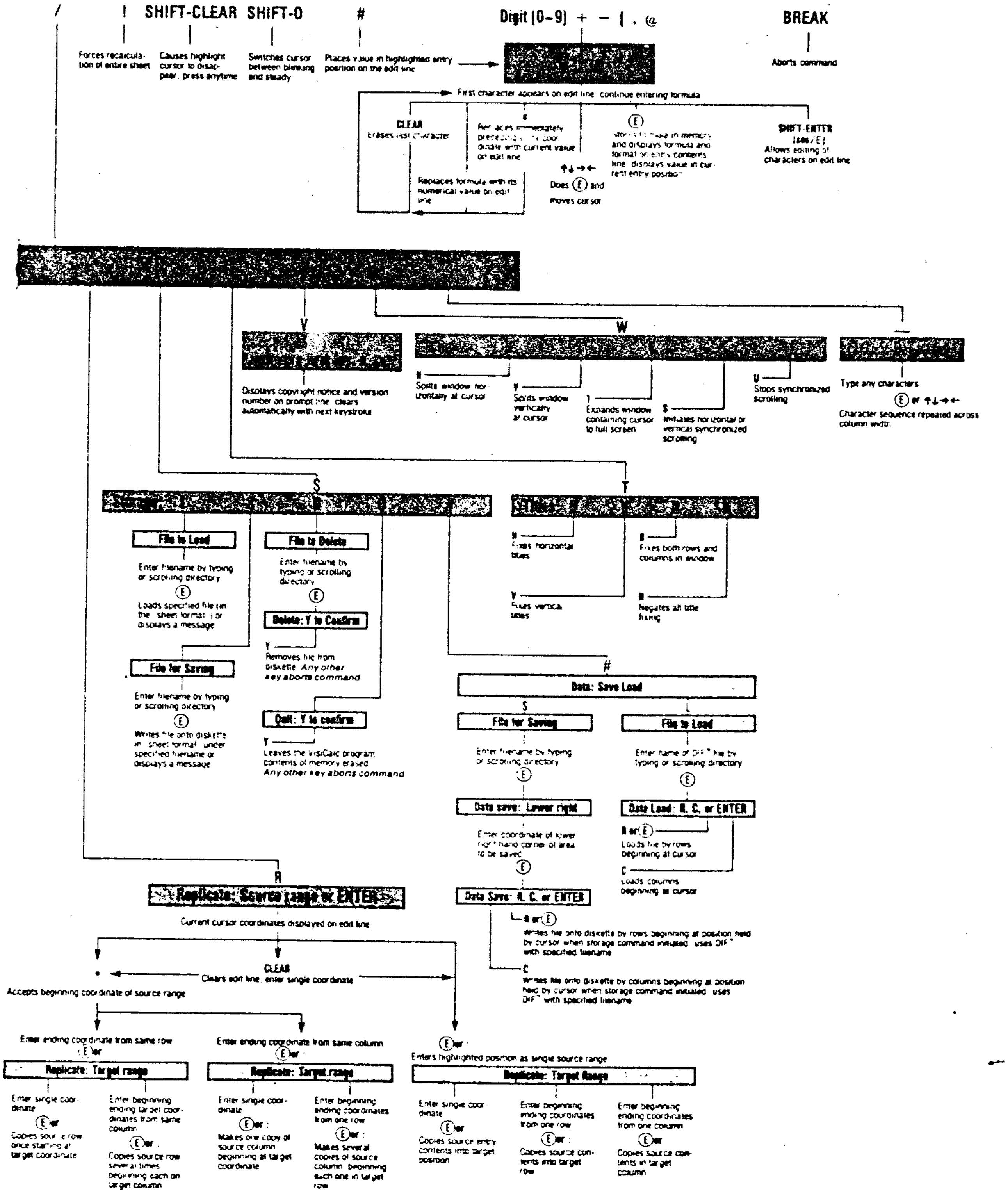
Blank removes only the label or value in the entry position on which the highlight cursor is located, but leaves the existing format setting for that position.

1) Move the highlight to the entry position to be erased.		
2) Type /	prompt line:	Command: BCDFGIMPRSTVW-
3) Type **B**	prompt line:	Blank
4) Press Ⓔ	The control panel will clear and the highlighted entry position will be blank.	
or press →, ←, ↑, or ↓	The control panel will clear, the original entry position will be blank, and the highlight will be on the next entry position.	

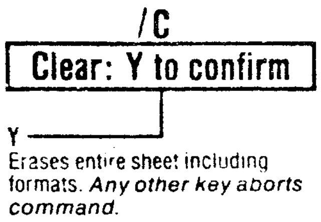
VISICALC COMMAND


>
Enter entry coordinate
Ⓔ
Moves cursor to specified entry position
:
Moves cursor between windows when split screen is in effect
→←↑↓
Moves cursor in direction of arrow
Any letter or "
Type any characters
CLEAR
Erases last character
SHIFT-ENTER
(see /E)
Allows editing of characters on edit line
↑↓→←
Does Ⓔ then moves cursor
/
B
Ⓔ or ↑↓→←
Erases contents of highlighted entry position, leaves formats. *Any other key aborts command*
C
Y
Erases entire sheet including formats. *Any other key aborts command*
D
R
Deletes all entries and formats in row containing cursor
C
Deletes all entries and formats in column containing cursor
E
E
Allows editing of entry contents of highlighted entry position
Ⓔ
Enters characters on edit line into entry position
CLEAR
Deletes character to left of edit cursor
any character
Inserts character to left of edit cursor
↑↓→←
Moves edit cursor without changing entry
R
Inserts row above cursor
C
Inserts column left of cursor.
M
Enter second coordinate within same column
Ⓔ
Row containing first coordinate moves to row above second coordinate
Enter second coordinate within same row
Ⓔ
Column containing first coordinate moves to left of column containing second coordinate
F
Format: D G I L R $ *
D
Defaults to global format of current window
G
General format
I
Integer display
L
Left justifies entry
R
Right justifies entry
$
Dollars and cents displays
*
Displays integer value as stars for bar graphs
G
Global: C O R F
Column Width
Type a number between 3 and maximum screen width; changes width of all columns to the number specified
Recalc Order: R C
R
Sets recalculation by rows (A1 to BK1, then A2 to BK2, etc.).
C
Sets recalculation by columns (A1 to A254, B1 to B254, etc.)
Recalc: A M
A
Recalculation proceeds automatically over whole sheet
M
Recalculation of entire sheet must be manually commanded by pressing !
Format: D G I L R $ *
(see /F)
Changes all unformatted entries in current window to specified format
P
Print: File Printer RS-232
F
Print: File name
Enter file name by typing or scrolling directory
Ⓔ
Writes exact copy onto diskette under displayed name in file format
P
R
Print: Lower right, Setup, &, or —
Enter coordinate of lower right corner of area to be printed
Ⓔ
Sends appropriate characters to printer, disk, or RS-232 device
Print: Setup or ENTER
Type appropriate setup string for non standard printers
Ⓔ
Restores line feed with each carriage return
Turns off line feed

STRUCTURE CHART


/
!
SHIFT-CLEAR
SHIFT-0
#
Digit (0–9) + – (. @
BREAK
Forces recalculation of entire sheet
Causes highlight cursor to disappear, press anytime
Switches cursor between blinking and steady
Places value in highlighted entry position on the edit line
Aborts command
First character appears on edit line, continue entering formula
CLEAR
Erases last character
#
Replaces immediately preceding coordinate with current value on edit line
Replaces formula with its numerical value on edit line
↑↓→←
Does Ⓔ and moves cursor
Ⓔ
Stores formula in memory and displays formula and format on entry contents line, displays value in current entry position
SHIFT-ENTER
Allows editing of characters on edit line
V
Displays copyright notice and version number on prompt line, clears automatically with next keystroke
W
H
Splits window horizontally at cursor
V
Splits window vertically at cursor
1
Expands window containing cursor to full screen
S
Initiates horizontal or vertical synchronized scrolling
U
Stops synchronized scrolling
–
Type any characters
Ⓔ or ↑↓→←
Character sequence repeated across column width
S
File to Load
Enter filename by typing or scrolling directory
Ⓔ
Loads specified file (in the sheet format) or displays a message
File for Saving
Enter filename by typing or scrolling directory
Ⓔ
Writes file onto diskette in sheet format under specified filename or displays a message
File to Delete
Enter filename by typing or scrolling directory
Ⓔ
Delete: Y to Confirm
Y
Removes file from diskette. Any other key aborts command
Quit: Y to confirm
Y
Leaves the VisiCalc program, contents of memory erased. Any other key aborts command
T
H
Fixes horizontal titles
V
Fixes vertical titles
B
Fixes both rows and columns in window
N
Negates all title fixing
#
Data: Save Load
S
File for Saving
Enter filename by typing or scrolling directory
Ⓔ
Data save: Lower right
Enter coordinate of lower right hand corner of area to be saved
Ⓔ
Data Save: R, C, or ENTER
R or Ⓔ
Writes file onto diskette by rows beginning at position held by cursor when storage command initiated, uses DIF™ with specified filename
C
Writes file onto diskette by columns beginning at position held by cursor when storage command initiated, uses DIF™ with specified filename
L
File to Load
Enter name of DIF™ file by typing or scrolling directory
Ⓔ
Data Load: R, C, or ENTER
R or Ⓔ
Loads file by rows beginning at cursor
C
Loads columns beginning at cursor
R
Replicate: Source range or ENTER
Current cursor coordinates displayed on edit line
.
Accepts beginning coordinate of source range
CLEAR
Clears edit line, enter single coordinate
Enter ending coordinate from same row
Ⓔ or
Replicate: Target range
Enter single coordinate
Ⓔ or
Copies source row once starting at target coordinate
Enter beginning ending target coordinates from same column
Ⓔ or
Copies source row several times beginning each on target column
Enter ending coordinate from same column
Ⓔ or
Replicate: Target range
Enter single coordinate
Ⓔ or
Makes one copy of source column beginning at target coordinate
Enter beginning ending coordinates from one row
Ⓔ or
Makes several copies of source column, beginning each one in target row
Ⓔ or
Enters highlighted position as single source range
Replicate: Target Range
Enter single coordinate
Ⓔ or
Copies source entry contents into target position
Enter beginning ending coordinates from one row
Ⓔ or
Copies source contents into target row
Enter beginning ending coordinates from one column
Ⓔ or
Copies source contents in target column

The CLEAR Command



Erases entire sheet including formats. *Any other key aborts command.*

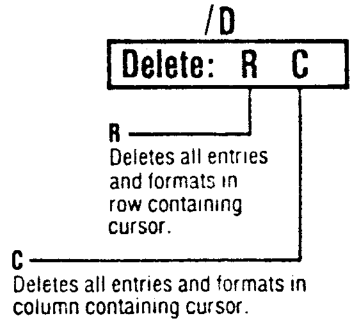
CLEAR sets all entry positions to blank, and resets formats, titles, windows, and other information about the sheet to the initial specifications set by the VisiCalc program when you first load it into the computer. The entry position highlight is returned to entry position A1.

Use this command to start with an empty VisiCalc sheet. Before using the clear command, be sure you have saved the sheet (see the STORAGE Command) if you do not want to lose the information you had written on it. All information erased with the clear command is irretrievable.

Example

1) Type /C	prompt line:	Clear: Type Y to confirm
2) Press Y	The screen will darken for a few seconds, then display the copyright and version notices. Your next keystroke will clear the prompt line.	

The DELETE Command



R: Deletes all entries and formats in row containing cursor.

C: Deletes all entries and formats in column containing cursor.

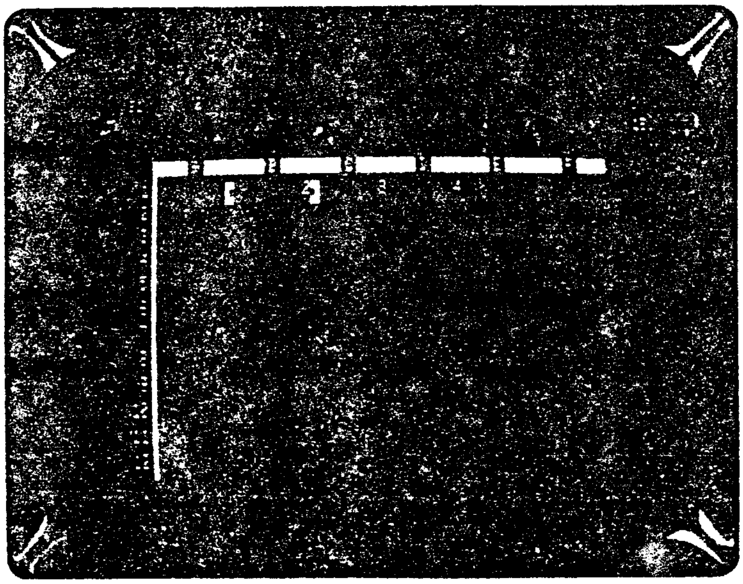
The VisiCalc program allows you to remove the entries from an entire column or row.

To use this command, first place the highlight in the row or column you want to delete. Then type **/D** The prompt line reads Delete: R C. Type **R** to delete the row in which the highlight is located or type C to delete the column. The VisiCalc program immediately deletes the information and moves all other rows or columns up to fill in the empty space.

The VisiCalc program also automatically looks up all value references (see the VALUE ENTRY Command) in formulas and changes them to correspond to the new coordinates resulting when the rows or columns are moved up. It then recalculates the whole sheet.

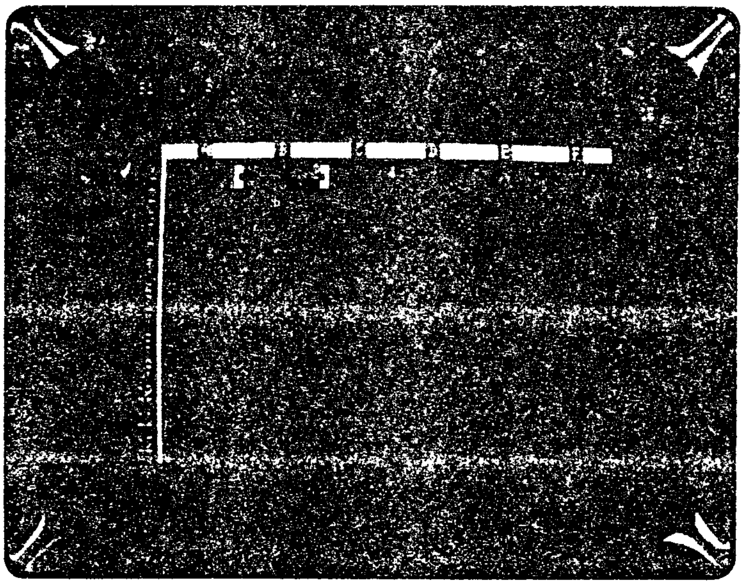
Example

1) Type **/CY**	To clear the sheet. Highlight is at A1.
2) Type **1→2→3→1+C1** Ⓔ	
3) Type **>B1** Ⓔ	Your screen should resemble this:



4) Type **/D** prompt line: Delete: R C

5) Type **C** Your screen resembles this:



6) Press ▶ entry contents line: C1 (V) 1+ B1

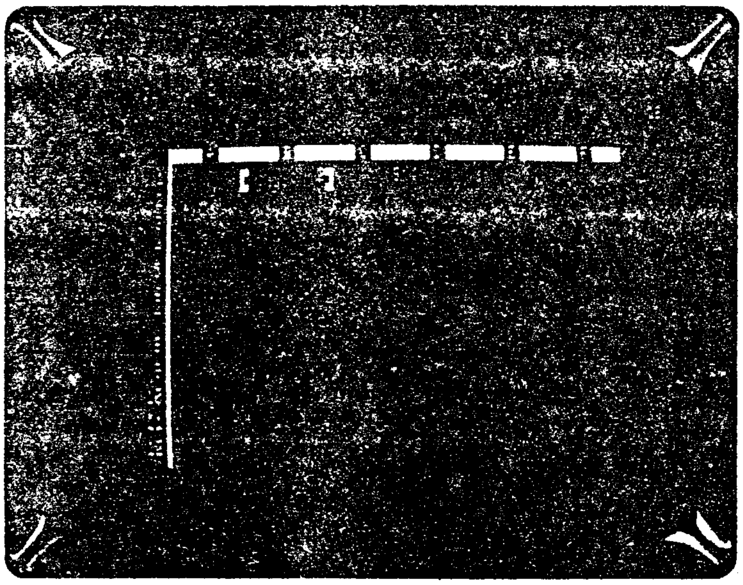
Note that the formula has been changed from 1+C1 to 1+ B1 so that it still refers to the value 3, which you originally entered at C1 in step 2 of this example. When the entries in column B were deleted, the VisiCalc program moved all columns up, and then changed the value reference so that the formula is correct with the new positions.

Use the delete command with care: You cannot recover the information which was recorded in a deleted row or column. The insert commands (see the INSERT Command) can recreate the row or column space, but not the actual entries.

If you delete an entry which you reference in a formula in some other part of the sheet (see the VALUE ENTRY Command), all entry positions containing formulas with value references to that position will contain the message ERROR.

Example

1) Type **/CY**	To clear the sheet. Highlight is at A1.
2) Type **1▶2▶1+B1◀**	The cursor is highlighting the 2 at B1. Your screen should resemble this:



3) Type **/D**

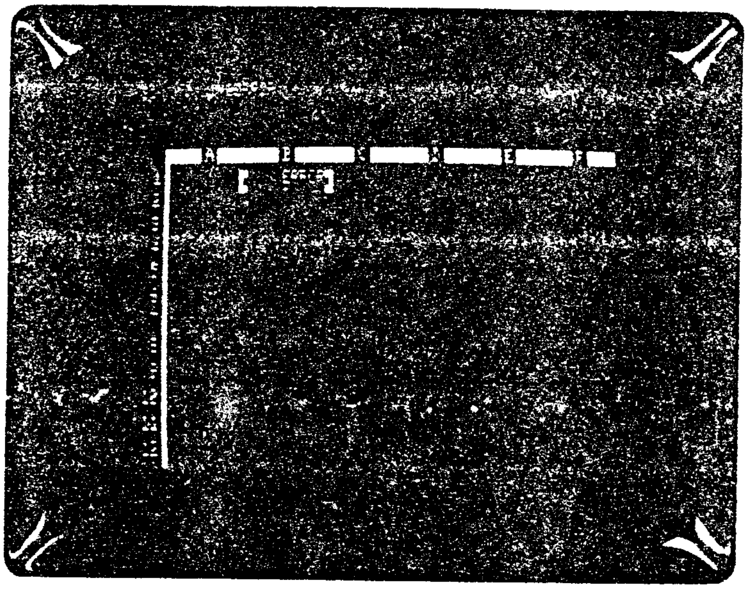
prompt line: Delete: R C

4) Type **C**

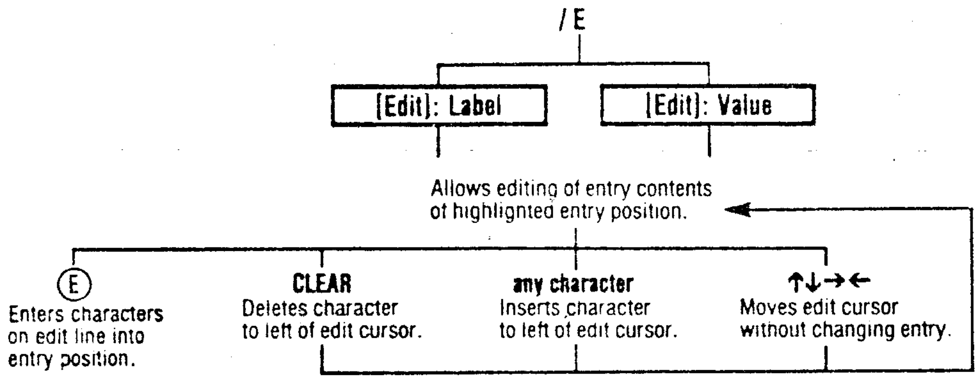
entry contents line: B1 (V) 1+ @ERROR

position B: ERROR

All information in column B was deleted and the contents of column C moved up. At step 2 in the example, you entered a formula with a value reference, 1+B1. The VisiCalc program replaced the B1 with @ERROR and alerts you by placing ERROR in the entry position containing the formula.



The EDIT Command


/E
[Edit]: Label
[Edit]: Value
Allows editing of entry contents of highlighted entry position.
Ⓔ
Enters characters on edit line into entry position.
CLEAR
Deletes character to left of edit cursor.
any character
Inserts character to left of edit cursor.
↑↓→←
Moves edit cursor without changing entry.

The Edit Command allows you to modify labels or formulas without typing the entire value, formula or word over again. You may delete and insert characters at any position in the entry. This is especially helpful when you have a long or complex formula which needs to be changed.

You can use the Edit Command either to edit an entry that is already part of the worksheet, or to edit something you are currently entering on the edit line.

To edit an entry on the edit line use the following:

1) Type **SHIFT–ENTER** (hold down **SHIFT** while pressing **ENTER**). The prompt line will read: [Edit]: Value or [Edit]: Label depending on the entry type. The edit cursor remains where it was at the end of the entry.

Continue with steps 3–6 below.

To edit an entry you have already entered onto the worksheet use the following steps:

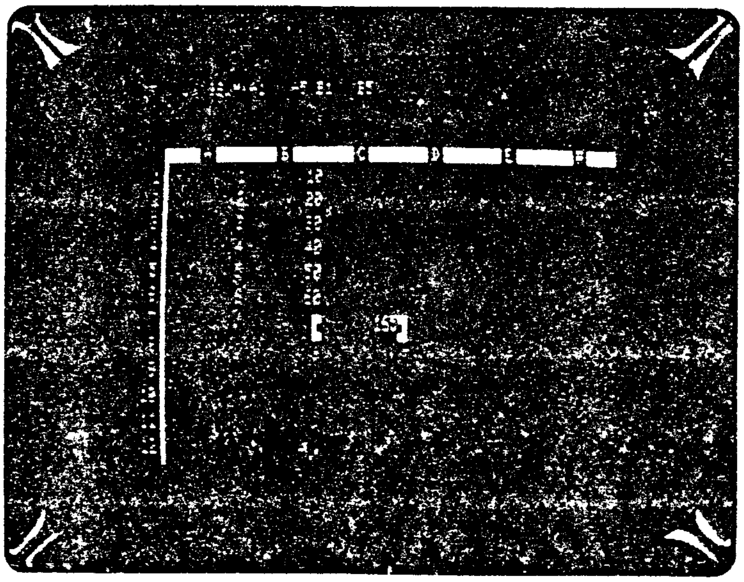
1) Place the cursor on the entry position with the entry you wish to change.
2) Type /E The prompt line will read: [Edit]: Value or [Edit]: Label depending on the entry type. The entry from the entry line is displayed on the edit line with the edit cursor on the first character.
3) Move the edit cursor with the arrow keys until it is one character to the right of the character(s) you wish to change. In the Edit Command the arrow keys move the edit cursor over the entry without changing it. The → and ← keys move the edit cursor to the right and left. The ↑ moves the edit cursor to the beginning of the entry. The ↓ moves it to the end of the entry.
4) Delete characters to the left of the edit cue by pressing the **CLEAR** key.
5) Insert characters to the left of the edit cursor by typing them in.
6) Press Ⓔ The prompt and edit lines clear, the new entry appears on the entry line and the label or value at the highlighted coordinate is changed accordingly.

Example

1) Type **/CY**	Clears the screen and places the cursor at A1	
2) Type **MISTEAK**	entry line: editline: The edit cursor follows the K.	Label MISTEAK
3) Type **SHIFT–ENTER**	prompt line: edit line: The edit cursor follows the K.	[Edit]: Label MISTEAK
4) Press ← twice.	edit line: The edit cursor highlights the A.	MISTEAK
5) Press **CLEAR**	edit line: The edit cursor still highlights the A.	MISTAK
6) Press → twice.	edit line: The edit cursor follows the K.	MISTAK
7) Type **E**Ⓔ	entry line: prompt line: edit line: A1:	A1 (L) MISTAKE clear clear MISTAKE

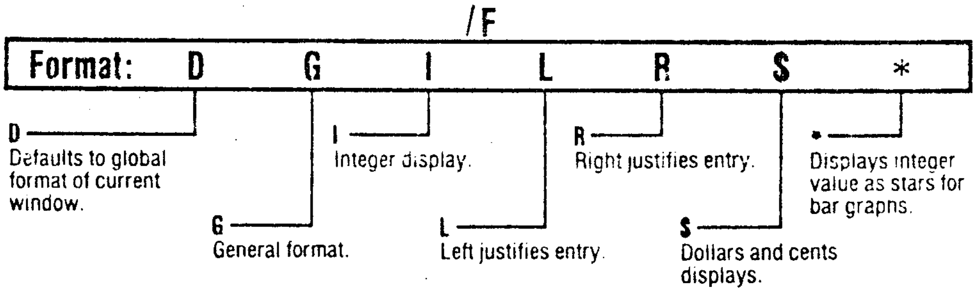
Example

1) Type **/CY**	Clears the screen and places the cursor at A1.
2) Type **1↓2↓3↓** **4↓5↓6**Ⓔ **>B1**Ⓔ **10↓20↓30↓40↓** **50↓60**Ⓔ **>C7**Ⓔ **@SUM(A1.A5,B1.B5)**Ⓔ	Your screen should resemble the photo below.



3) Type /E	prompt line:	[Edit]: Value
	edit line:	@SUM(A1 . . . A5, B1 . . . B5)
	The edit cursor is over the @.	
	C7:	165
4) Press ▶ twelve times	edit line:	@SUM(A1 . . . A5, B1 . . . B5)
	The edit cursor is over the comma following A5.	
5) Press **CLEAR**	edit line:	@SUM(A1 . . . A, B1 . . . B5)
	The edit cursor remains over the comma.	
6) Type **6**	entry line:	@SUM(A1 . . . A5, B1 . . . B5)
	edit line:	@SUM(A1 . . . A6, B1 . . . B5)
7) Press ▶ eight times	Edit cursor moves to cover the).	
8) Press **CLEAR** and type **6**	entry line:	C7 (V) @SUM(A1 . . . A5, B1 . . . B5)
	edit line:	@SUM(A1 . . . A6, B1 . . . B6)
9) Press Ⓔ	entry line:	@SUM(A1 . . . A6, B1 . . . B6)
	prompt line:	clear
	edit line:	clear
	C7:	231

The FORMAT Command


/F
Format: D G I L R $ *
D
Defaults to global format of current window.
G
General format.
I
Integer display.
L
Left justifies entry.
R
Right justifies entry.
$
Dollars and cents displays.
*
Displays integer value as stars for bar graphs.

The format command adds an explicit display condition to the highlighted entry on the VisiCalc sheet. The original entry contents (LABEL or VALUE) remain unchanged in the computer's memory. They are always used in calculations and are displayed completely on the entry contents line. On the sheet, however, the entry will appear as formatted by this command. If the entry is moved to another position on the sheet (see the REPLICATE Command, the MOVE Command, and the INSERT Command), printed on paper or another device (see the PRINT Command), or stored on diskette (see the STORAGE Command) its format stays with it.

Once an entry position has been formatted, a format indicator will appear on the entry contents line between the current entry coordinates and entry contents. Erasing the contents of the entry position (see the BLANK Command) does not remove explicit formats, but clearing the sheet (see the CLEAR Command) does.

To format a row or a column before making entries, format the first entry and then use the replicate command (see the REPLICATE Command) to copy the format into the rest of the row or column.

It is not possible to change the number of character spaces in an individual entry position. To change the number of characters in every column throughout the window, use the global command for changing column width (see the GLOBAL Command).

To see how the format options affect the display, set up your sheet as instructed in the example below. All the examples in this section use this sheet for the format examples.

Example

1) Type /CY	This clears the sheet, positions the cursor at A1, and resets all display characteristics.
2) Type **LABEL ENTRY** **⬍1.23456789⬍99.999Ⓔ**	

You now have one label and two values on your sheet. Any explicit format settings were removed when the sheet was cleared. You have one window (see the WINDOW Command), no titles (see the TITLE Command), your columns contain nine character spaces, which is the general column width (see the GLOBAL Command), and all entry positions display the general format (see the GLOBAL Command and the discussion below). These same display characteristics are set each time the VisiCalc program is loaded into your computer and whenever the clear command is used.

The Default Display Format—/FD

An individual entry position format is changed to whatever format has been previously set with the global command (see the GLOBAL Command). If no global setting is in effect, the default will be the general format, described in /FG below.

Example

1) Type **>C1Ⓔ**

2) Type **/F$**

The highlighted entry (C1) changes to dollars and cents format; B1 is unchanged. The entry contents line displays the /F setting.
entry contents line: C1 /F$ (V) 99.999
position C1: 100.00

3) Type **/FD**

entry contents line: C1 (V) 99.999
position C1: 99.999
The default setting is general, as set by /CY.

4) Type **/GFI**

All values in the window which have **not** received an individual format are displayed as integers. The global setting is now integer format (see the GLOBAL Command).
position B1: 1
position C1: 100

5) Type **/FG**

entry contents line: C1 /FG (V) 99.999
position B1: remains in integer format
position C1: 99.999
The setting on C1 is now general, as shown on the entry contents line.

6) Type **/FD**

entry contents line: C1 (V) 99.999
position B1: unchanged
position C1: 100
The setting on C1 has been removed and is no longer indicated on the entry contents line. However, since there is a global integer setting in effect, position C1 is displayed in integer format instead of general format.

The General Format—/FG

Labels begin in the leftmost space (left justified) and are cut off wherever the column width ends.

Values are moved as far to the right as possible (right justified) with a leading blank character in the leftmost position in the column. Decimal or scientific notation is selected to display the largest number of significant digits.

Example

1) Type **>A1**Ⓔ	Position A1 is now in general format for labels. entry contents line: A1 (L) LABEL ENTRY position A1: LABEL ENT
2) Press →	position B1: 1 A global integer setting is still in effect from the previous example.
3) Type **/FG**	position B1: 1.234568 General format for numbers with few significant digits.
4) Type **123456789123456789**Ⓔ	entry contents: 123456789123000000 position B1: 1.235E17 General format for numbers with many significant digits. The VisiCalc program selects scientific notation when this permits a larger number of significant digits to be displayed within the current column width.

The Integer Format—/FI

All values are displayed rounded to the nearest whole number.

Example

1) Type **1.2**Ⓔ	entry contents line: B1 /FG (V) 1.2 position B1: 1.2
2) Type **/FI**	entry contents line: B1 /FI (V) 1.2 position B1: 1 Entry contents are rounded to the nearest integer.

Left Justification—/FL

All labels begin in the leftmost character space; all values are moved to left, preceded by one leading blank. This command affects only the display of entries on the sheet which have fewer characters than the current column width.

Example

1) Type **/FL**	entry contents line:	B1 /FL (V) 1.2
	position B1:	1.2
	The value in B1 has been moved to the left.	

Right Justification—/FR

The last character of a label or a value falls in the last character space of the entry position. This command affects only the display of entries on the sheet which have fewer characters than the current column width.

Example

1) Type **/FR**	entry contents line:	B1 /FR (V) 1.2
	position B1:	1.2
	The value in B1 has been moved to the right.	

Dollars and Cents Display—/F$

All values are rounded to two decimal places. No $ is displayed in the entry position. Trailing zeros are shown. There is no effect on labels.

Example

1) Type **/F$**	entry contents line:	B1 /F$ (V) 1.2
	position B1:	1.20

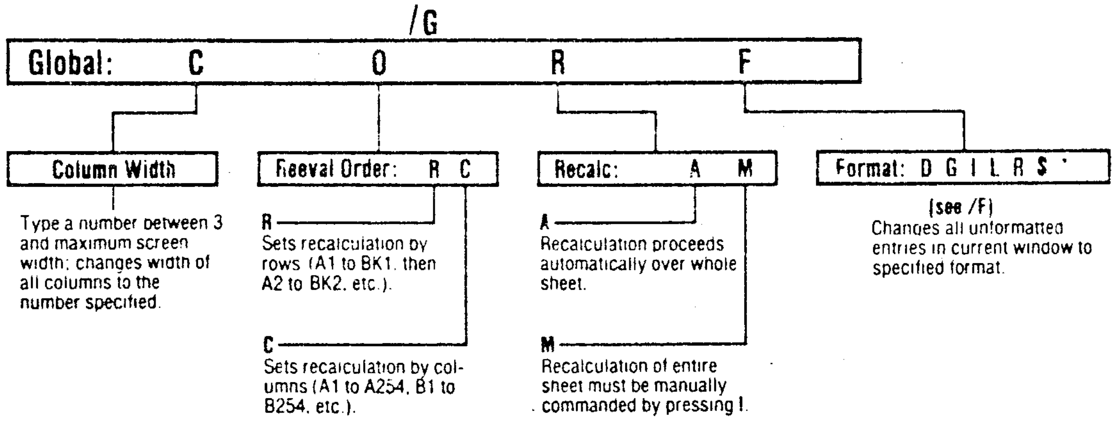
Bar Graph Format—/F*

All values are truncated to an integer. That integer is replaced in the entry position by an equal number of stars. If the number of stars is larger than the width of the entry position, extras are ignored.

Example

1) Type **/F***	entry contents line:	B1 /F* (V) 1.2
	position B1:	*

The GLOBAL Command



Type a number between 3 and maximum screen width; changes width of all columns to the number specified.

R
Sets recalculation by rows (A1 to BK1, then A2 to BK2, etc.).

C
Sets recalculation by columns (A1 to A254, B1 to B254, etc.).

A
Recalculation proceeds automatically over whole sheet.

M
Recalculation of entire sheet must be manually commanded by pressing !.

(see /F)
Changes all unformatted entries in current window to specified format.

The four global command options affect the window display and the order in which rows and columns are calculated.

The Global Column—/GC

This command allows you to change the width of all columns in a window to any number of characters between 3 and the maximum width of your window. Different column widths may be set within each window but may not be set individually. The VisiCalc program will display as many whole columns of a given width as possible in the window containing the cursor. When changing to a single, wide column in a window, any vertical title areas previously set will be automatically removed (see the TITLE Command).

Example

1) Type **/CY**	This clears the sheet and resets the column width to nine characters.	
2) Type in position A1: **THIS LINE IS TOO LONG.Ⓔ**	entry contents line:	THIS LINE IS TOO LONG.
3) Type **/G**	prompt line:	Global: C O R F
4) Type **C**	prompt line:	Column width:
5) Type **18** Ⓔ	entry contents line:	THIS LINE IS TOO LONG.
	position A1:	THIS LINE IS TOO L
6) Type **/GC24**Ⓔ	entry contents line:	THIS LINE IS TOO LONG.
	position A1:	THIS LINE IS TOO LONG.

When displaying numbers on the sheet, the VisiCalc program may change the format from the way you originally entered it on the edit line. Numbers will be rounded off in the window display when necessary to fit into the column width. The VisiCalc program will use scientific notation if this will permit more significant digits to be shown at the current column width than regular decimal notation. If the column is too narrow to display the integer portion of a value in either decimal or scientific notation, the VisiCalc program will display as many > symbols as will fit in the column width in the entry position with a leading blank.

Regardless of how the values appear on the screen, the numbers stored in the computer's memory will remain unchanged and will be used in all calculations. The number, as you originally entered it on the edit line, will be displayed on the entry contents line whenever the cursor is at its entry position.

Example

1) Type **/CY**

2) Type **123456789Ⓔ**	entry contents line: position A1:	A1 (V) 123456789 1.2346E8
	Note the leading blank, scientific notation, and final digit rounded up.	
3) Type **▶1.23456789Ⓔ**	entry contents line: position B1:	B1 (V) 1.23456789 1.234568
	Note the leading blank and the final digit rounded up.	
4) Type **/GC3Ⓔ**	entry contents line: position A1: position B1:	B1 (V) 1.23456789 >> 1.
	The number in A1 is too large to express with two characters, which is the space available for digits when the column width is 3, so VisiCalc displays >>. The number in B1 is rounded as shown.	

The Global Order of Recalculation—/GO

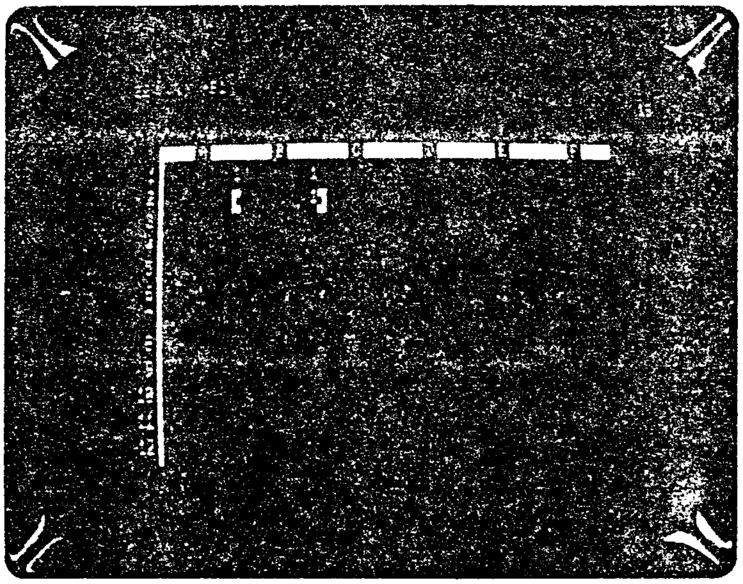
This command allows you to set the order of calculation to Columns or Rows. In columnwise calculation and recalculation, entry position A1 is evaluated, then A2, A3, to the bottom of the sheet, then B1, B2, B3, to the bottom of the sheet in this column, then C1, C2, and so on. In row calculation, A1 is first, then B1, C1, and on to the right hand end of the sheet, then A2, B2, C2 to the end of the sheet. This order of calculation is indicated on the control panel by the letter C or R in the upper right hand corner of the entry contents line. When you first load the VisiCalc program, the calculation order is by column.

If the VisiCalc program appears to evaluate formulas incorrectly, you have placed formulas in entry positions so that they are calculated before the value references that they contain. Order of calculation and recalculation has been discussed at length in the VALUE ENTRY Command in Part III, and also in Part II, Lesson Three, in the section entitled "The Order of Recalculation."

The following example illustrates the use of value references set up for column calculation, and shows what happens when they are calculated in a rowwise fashion.

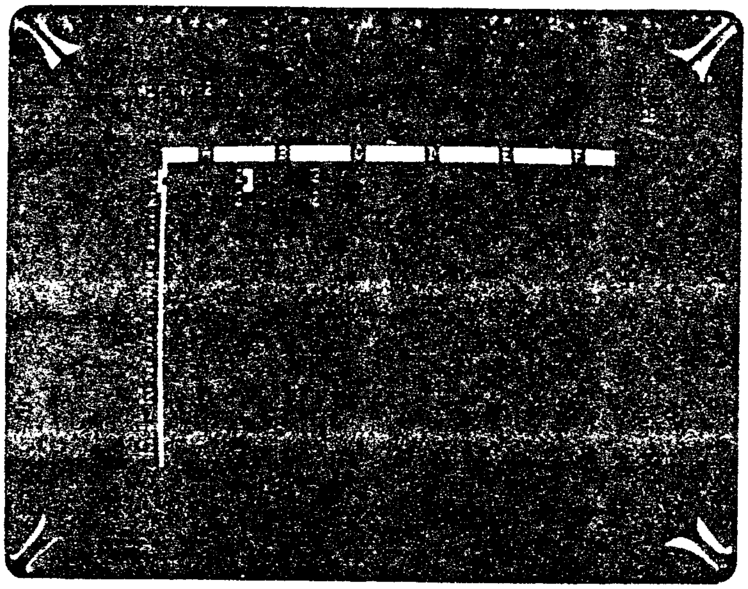
Example

1) Type **/CY**	This clears the sheet and sets column calculation.
2) Type 1↓	
3) Type **+A1**Ⓔ	
4) Type **>B1**Ⓔ	
5) Type **+A2**↓	
6) Type **+B1**Ⓔ	Your screen should resemble this:



7) Type >**A1**Ⓔ2Ⓔ

The value at A1 is changed to 2, and the other values are recalculated. Your screen should resemble this:

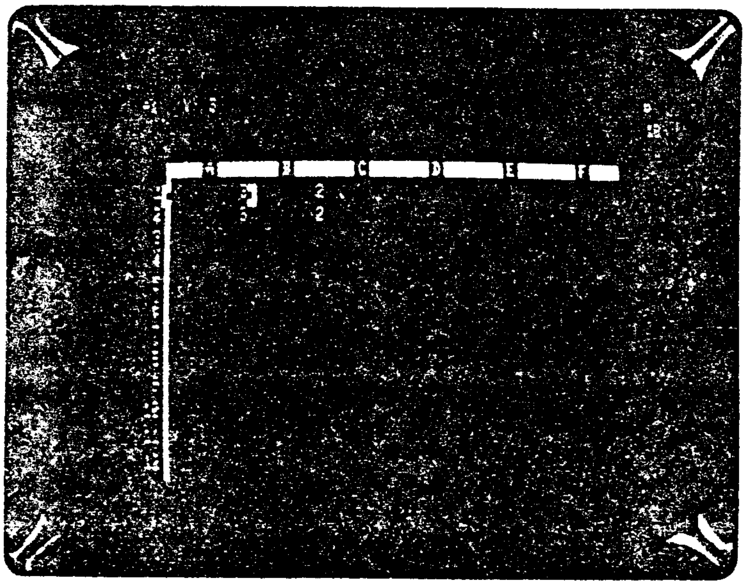


8) Type /**GOR**

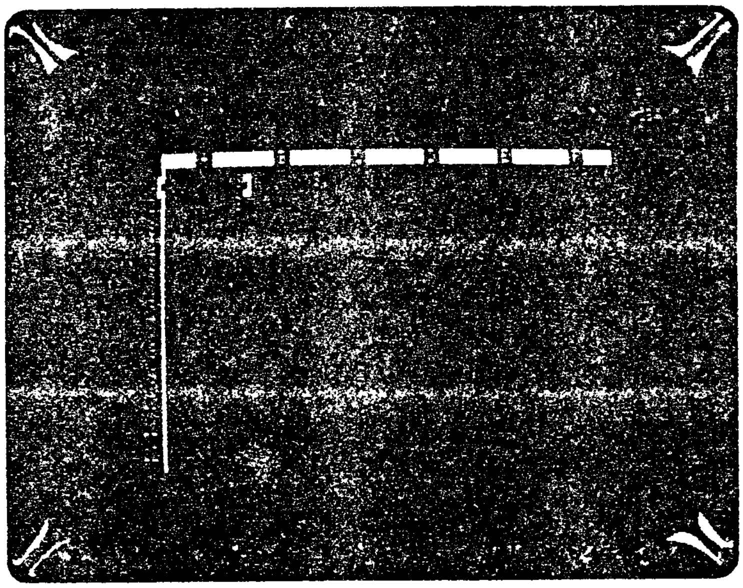
The order of recalculation indicator changes to R.

9) Type 3Ⓔ

Your screen resembles this:



This is incorrect. B1 should be the same as A2, yet it displays a different value because B1 was recalculated **before** A2 was recalculated. In such instances, you can force a recalculation by pressing ! when the prompt line is blank. Press ! now and the screen shows the correct values as in the photo below.



A similar example that works correctly for row order of calculation, but not for column order, is this (you can try this on your own):

1) Type **/CY** The sheet is cleared and calculation is set to column order.
2) Type **/GOR** To change to row order of calculation.
3) Type **1➧+A1Ⓔ**
4) Type **>A2Ⓔ**
5) Type **+B1➧+A2Ⓔ**

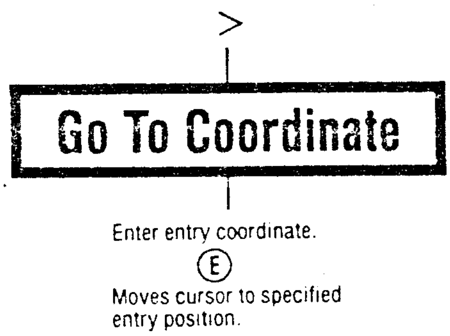
Automatic or Manual Recalculation—/GR

When first loaded, the VisiCalc program is set to recalculate all values as soon as any change is made in an entry position to which a formula may refer. Changes ripple rapidly across your sheet whenever you change a value. Sometimes, such as when you are typing in a column or row of figures, you would like to turn off recalculation completely to eliminate the pause that occurs when the VisiCalc program is recalculating. To turn it off, type **/GRM** which stands for GLOBAL:RECALCULATION MANUAL. Under manual recalculation, only the highlighted formula will be recalculated automatically. The whole sheet is updated only when you type ! There is no cue on the control panel to indicate whether you are in manual or automatic recalculation but the status should be evident from watching the behavior of the sheet. To resume automatic recalculation, type **/GRA**

The Global Format—/GF

This command allows you to assign a format setting for all entry positions in the window which have not been individually formatted. The global format command uses the same formatting options as are available for individual entry positions. See the FORMAT Command for an explanation of these options. (Global Format Default (/GFD) has no effect on the VisiCalc screen.) If you have split your screen into two windows (see the WINDOW Command), you may use a different global format for each window. When you save your sheet with /SS all global formats are also saved (see the STORAGE Command).

The GO TO Command



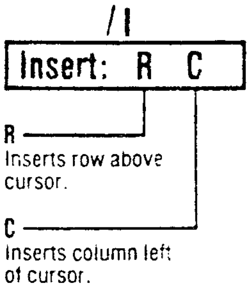
With this command, you have a fast way to move the highlight to any position on the sheet. To move the highlight cursor to a new entry position, type > followed by the coordinates of the entry position to which you want to jump, and then Ⓔ

Example

1)	Type >	prompt line: Go to: Coordinate
2)	Type **AM205** Ⓔ	The highlight will be moved to column AM, row 205.

Entering an invalid coordinate, such as AB300, causes the control panel and the column borders to flash, and the VisiCalc program to clear the prompt and edit lines.

The INSERT Command

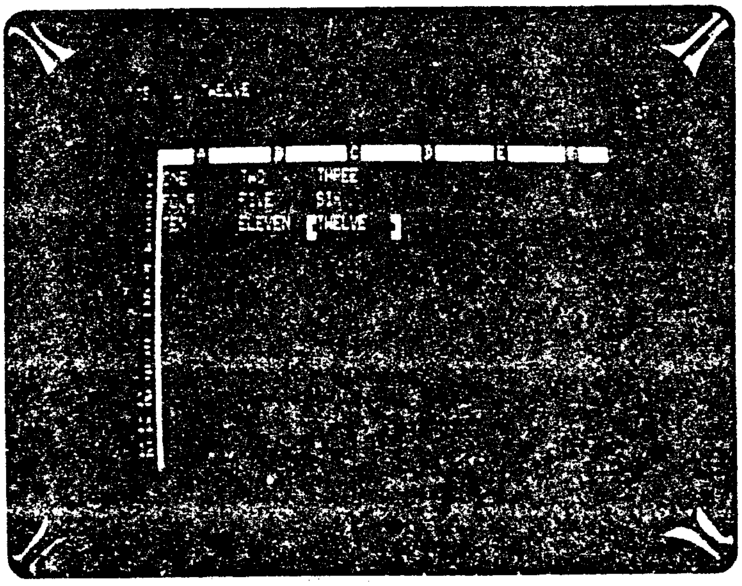


While developing a VisiCalc sheet, you may find that you need to insert additional rows or columns. The insert command gives you this ability. After you type /I VisiCalc expects you to enter R if you want a row inserted, or a C to insert a column. It is not possible to insert a single entry position—you may insert only an entire row or column. If there is data in row 254 you cannot insert a row. If there is data in column BK, you cannot insert a column. However you can use the MOVE command in these cases.

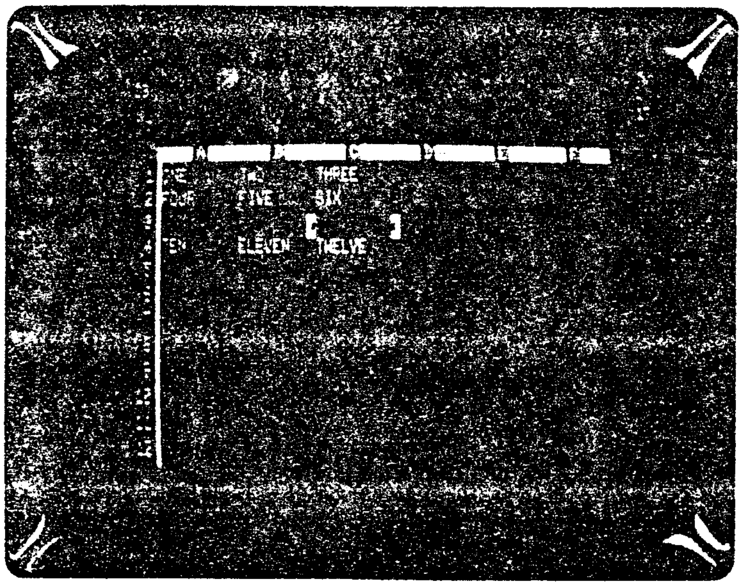
Inserting a New Row—/IR

This command inserts a new, blank row in the row in which the cursor is situated when you give the command. All rows that were at or below the cursor are moved down one row to make room for the insertion. The VisiCalc program then changes all value references in formulas to reflect the new entry position coordinates in the rows that were moved. For example, if a formula contains the coordinate C2, and a row is inserted at row 2, the VisiCalc program will change the coordinate to C3. The cursor will remain in its old position in the new blank row,

Example
Set up your screen as in the photo below.



Now type /IR Your screen should like the photo below.



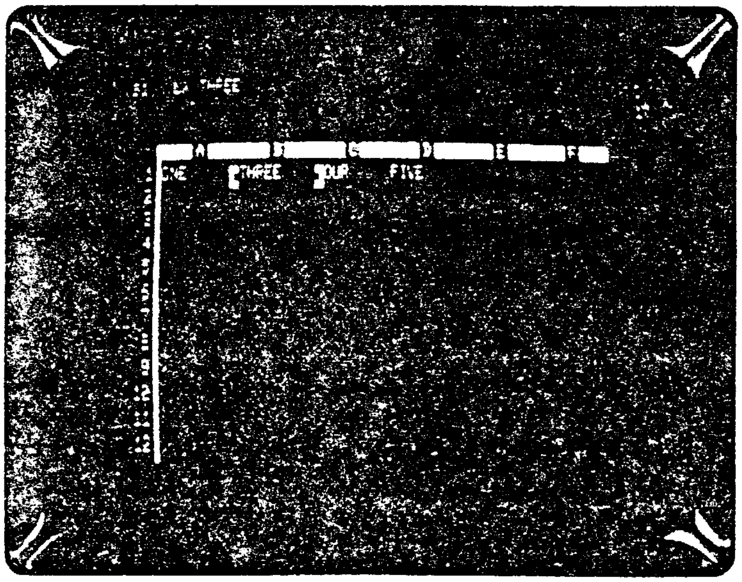
```
     A       B       C       D       E       F
 1  ONE     TWO     THREE
 2  FOUR    FIVE    SIX
 3
 4  TEN     ELEVEN  TWELVE
```

Inserting a New Column—/IC

This command inserts a new, blank column in the column in which the cursor is located when the command is given. All columns at and to the right of the cursor are moved one column to the right to make room. The VisiCalc program changes all value references in formulas to reflect the new entry position coordinates for the columns that were moved. The cursor remains in its original position in the new, blank column.

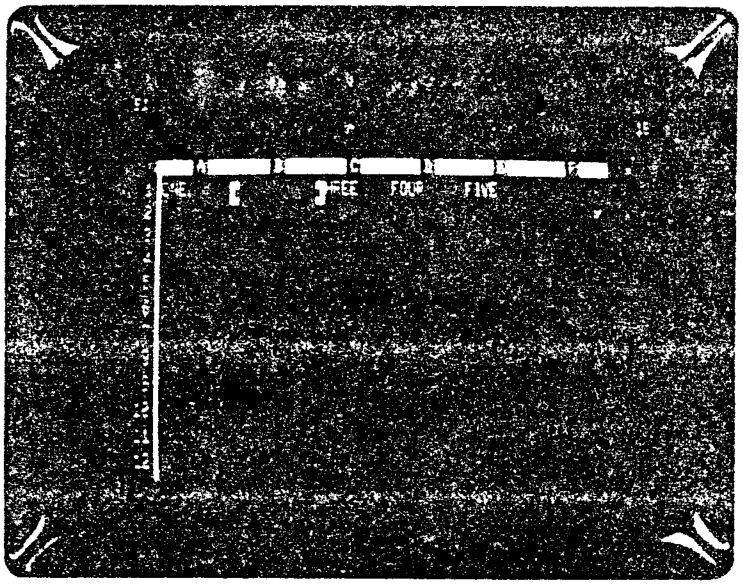
Example

Set up your screen as in the photo below.

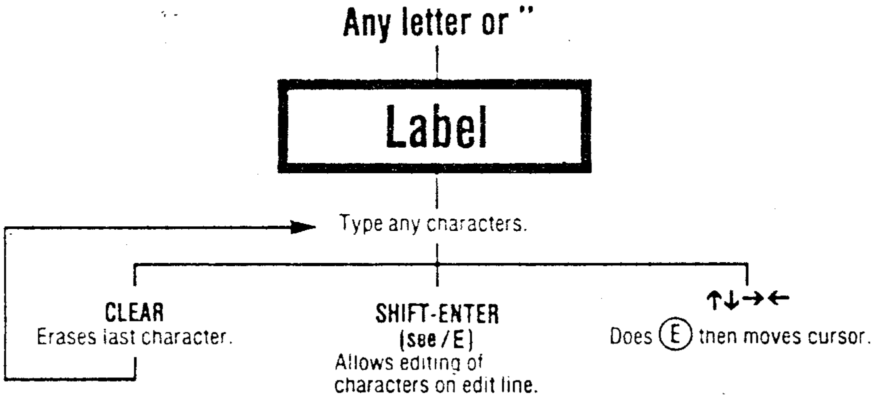


```
     A       B       C       D       E       F
 1  ONE     THREE   FOUR    FIVE
```

Now type /IC Your screen should look like the photo below.



The LABEL ENTRY Command



Any combination of alphanumeric characters can be put into an entry position. The character strings entered as labels are evaluated by the VisiCalc program as 0.

You begin the label entry command from a cleared control panel. The label will be entered in the entry position on which the cursor is sitting when you begin the label entry command. When the first character of the label is alphabetic, just begin typing the label. The VisiCalc program immediately recognizes that you are entering a label.

You may want to enter a label such as 2ND MONTH. In such a case where the first character is a digit or one of the arithmetic operators which automatically begin a value entry (see VALUE Command), you must type a quotation mark (") as the first character in the label. The " will not appear as part of the label.

As soon as you have pressed either the " or a letter, the prompt line will display Label and the characters of the label will appear in the edit line and in the entry position. Press **CLEAR** to back over any mistyped characters and retype them.

When the label is complete, terminate the command with either Ⓔ, →, ←, ↓ or ↑. The label will appear in the entry position with the first character at the left side of the column and will have replaced anything that you may have previously placed in that position. You may tell the VisiCalc program to display the label right-justified within the column with a special format (see the discussion of /FR in Part III, the FORMAT Command). All entries made with the label command are given the entry type L, which appears on the entry contents line as (L).

Example

1) Type **>A1**Ⓔ	The cursor is in entry position A1.	
2) Type **P**	prompt line:	Label
	edit line:	P
3) Type **ERIOD**	entry contents line:	A1
	prompt line:	Label
	edit line:	PERIOD
4) Press Ⓔ	entry contents line:	A1 (L) PERIOD
	prompt line:	clear
	edit line:	clear
or press ↓, →, ↑, or ←	The label is entered and highlight is on the next entry position.	

Suppose you want to use a label that looks like a formula.

Example

1) Type **>B1**Ⓔ		
2) Type **"**	prompt line:	Label
3) Type **.575*B2**	prompt line:	Label
	edit line:	.575*B2
4) Press Ⓔ, →, ←, ↓ or ↑	entry contents line:	B1 (L) .575*B2
	prompt line:	clear
	edit line:	clear

The (L) indicates entry type and allows you to distinguish the label entry from a value entry.

The MOVE Command

/M

Move: From...To

Enter second coordinate within same column. Ⓔ Row containing first coordinate moves to row above second coordinate.	Enter second coordinate within same row. Ⓔ Column containing first coordinate moves to left of column containing second coordinate.

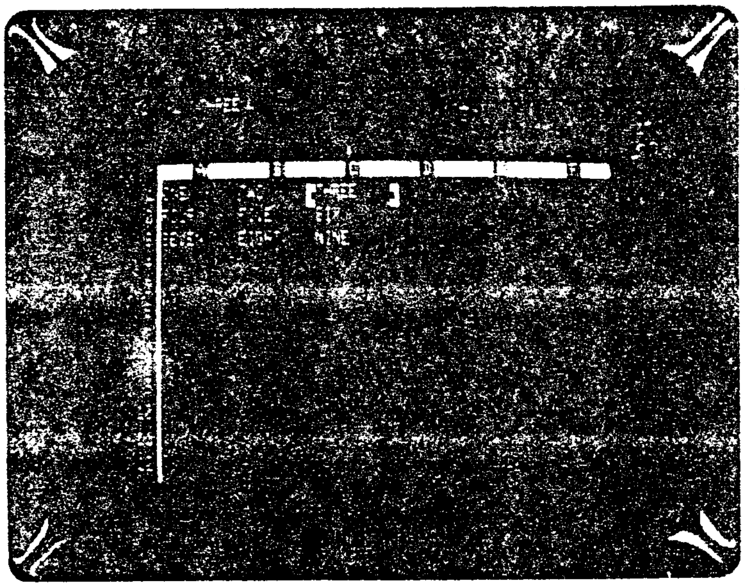
It is sometimes desirable to change the position of a row or a column. The move command lets you do just that. The VisiCalc program will automatically change all value references in formulas (see the VALUE ENTRY Command) to reflect the new coordinates which result when the rows or columns are moved. When you move a row or a column, be careful that you don't move formulas containing value references to positions where the formulas are calculated before the value references. Order of calculation and recalculation is discussed at length in the VALUE ENTRY Command in Part III and also in Part II, Lesson Three, in the sections entitled "The Order of Recalculation" and "Forward and Circular References."

To Move a Row

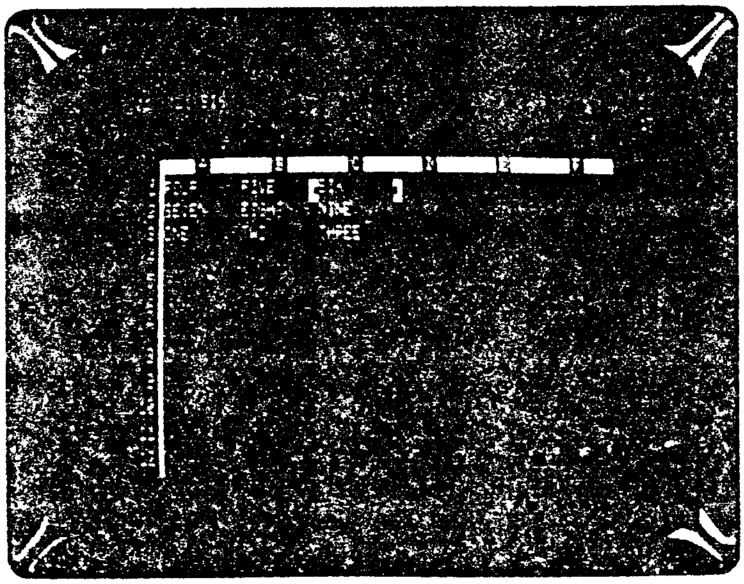
When moving a row to a lower position on the sheet, row 1 to row 3 for instance, first place the cursor on the row you wish to move. Then type /M The coordinate of the highlighted entry position will appear on the edit line. Press . and then use the cursor-moving keys to place the cursor on the row that is just below the row to which you want to move. In other words, the row will be moved to the position just above the cursor. When moving rows, the cursor may be in any position within the row when you initiate the command; however, when you indicate the new location, you must use the same column coordinate. For instance, moving C10 to C5 is valid, but moving C10 to D5 is not. Terminate the command with Ⓔ When a row is moved, all intervening rows and columns are moved up to fill in the place vacated by the moved row or column. The VisiCalc program then changes all value references to reflect the new position.

Example

1) Set up your sheet as in the photo:



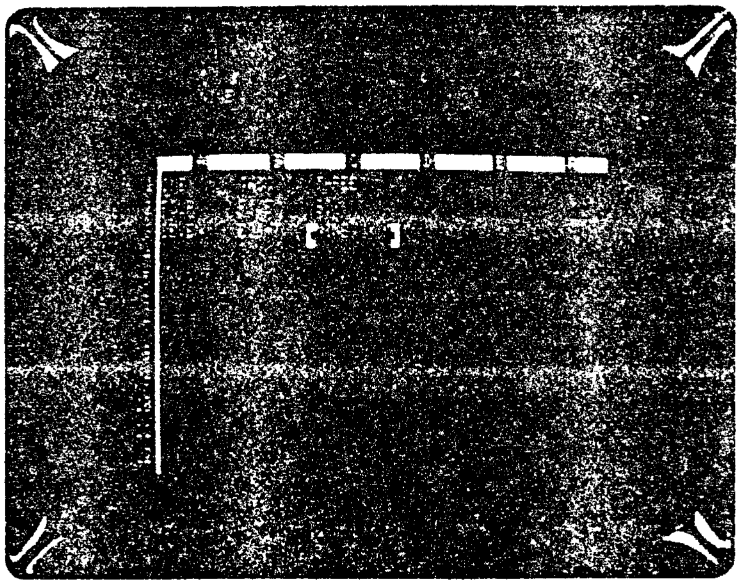
2) Type **/M**	prompt line: edit line:	Move: From . . . To C1
3) Type **.**	edit line: Row 1 is going to be moved.	C1 . . .
4) Press →→→	edit line: Row 1 is going to move to row 3.	C1 . . . C4
5) Press Ⓔ	The sheet should resemble this photo:	



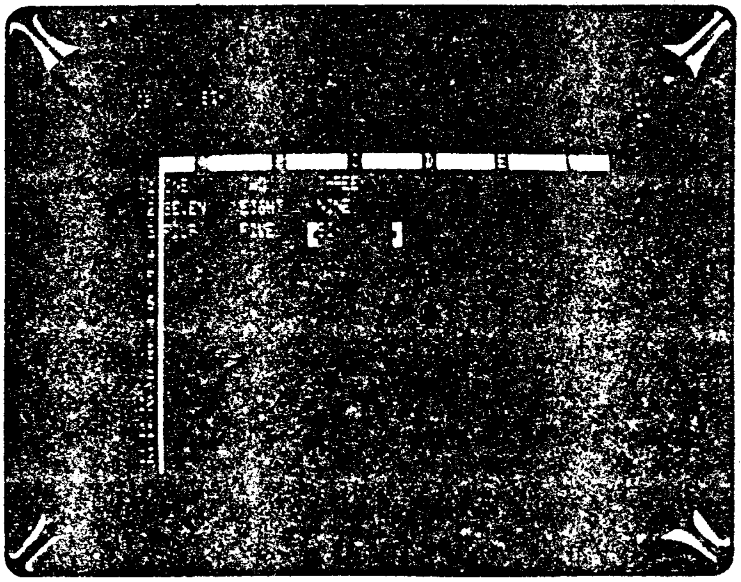
When you move a row to a higher place on the sheet, row 3 to row 2 for instance, either point with the cursor or type in the coordinate of the target row.

Example

1) Set up your sheet as in this photo:



2) Type /M	prompt line: edit line:	Move: From . . . To C3
3) Type .	edit line: Row 3 will be moved.	C3 . . .
4) Press ↑	edit line:	C3 . . . C2
5) Press [illegible]	Row 3 has moved to row 2.	

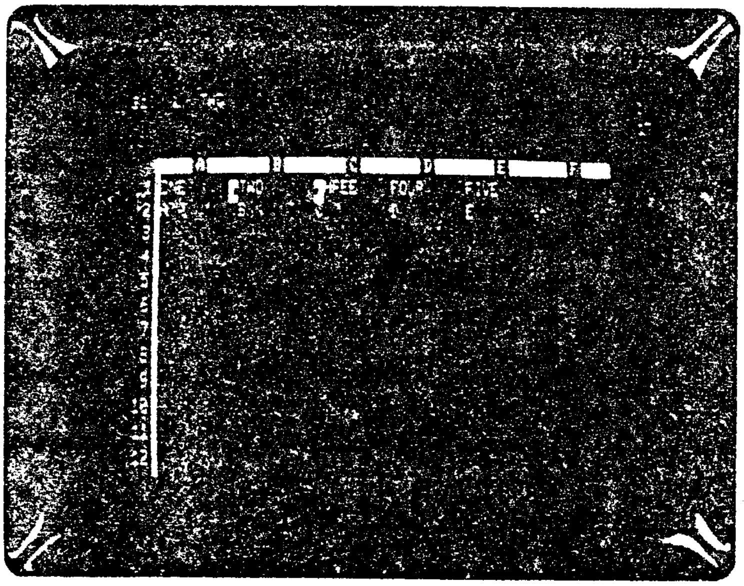


To Move a Column

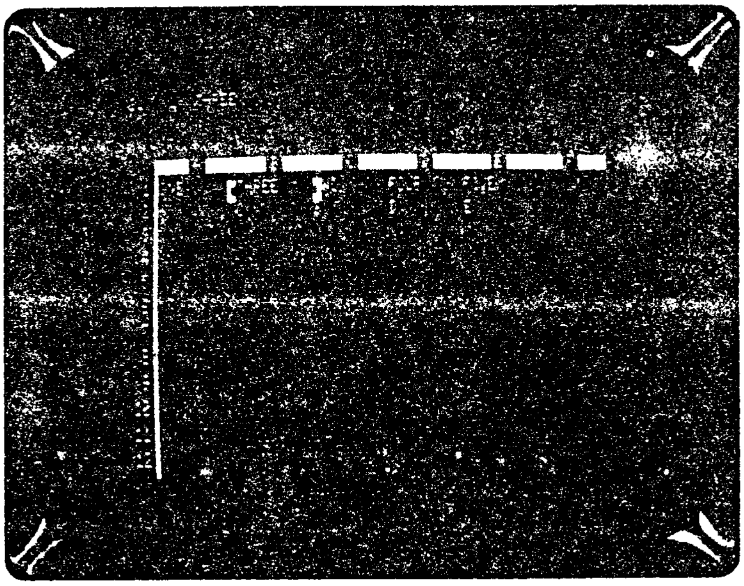
To move a column to the right of its current position, first place the cursor on the column you wish to move. Type /M and the coordinate of the highlighted entry position will appear on the screen. Press . and then point with the cursor (or type in the coordinate) to the column just to the right of the desired new location. The column will be moved to the position just to the left of the cursor. When moving columns, the cursor can be in any position within the column when you start the command, but when you indicate the new location, you must use the same row coordinate. For example, moving A2 to D2 is valid, but moving A2 to D1 is not. End the command with Ⓔ. All intervening columns are moved up to fill the place vacated by the moved column.

Example

1) Set up your sheet like this, with the highlight on B1:



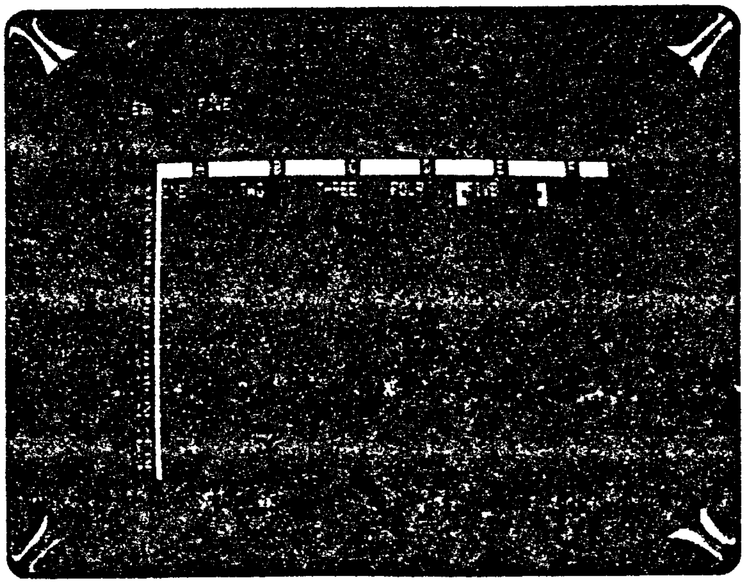
2) Type /M	prompt line:	Move: From . . . To
	edit line:	B1
3) Press .	edit line:	B1 . . .
4) Press D1	edit line:	B1 . . . D1
5) Press Ⓔ	Column B has moved to column C as in the photo:	



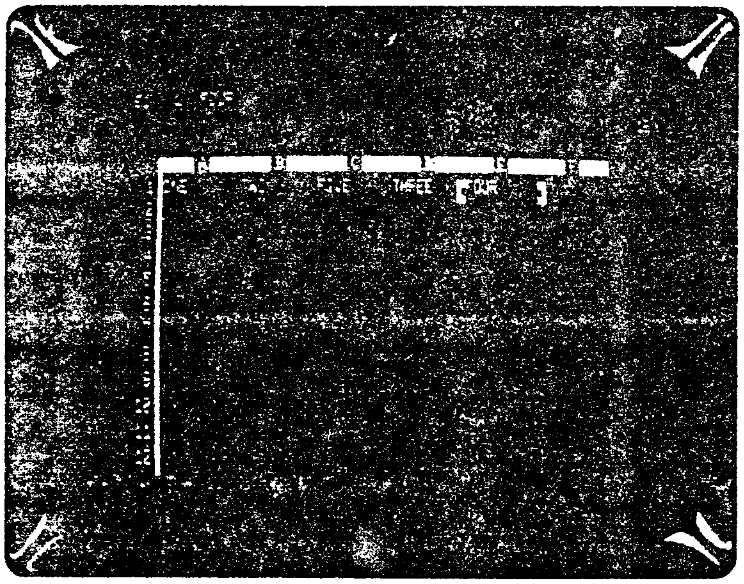
When you move a column to the left of its present position, either point with the cursor or type in the coordinate of the target row.

Example

1) Set up your sheet like this photo, with the cursor on E1:



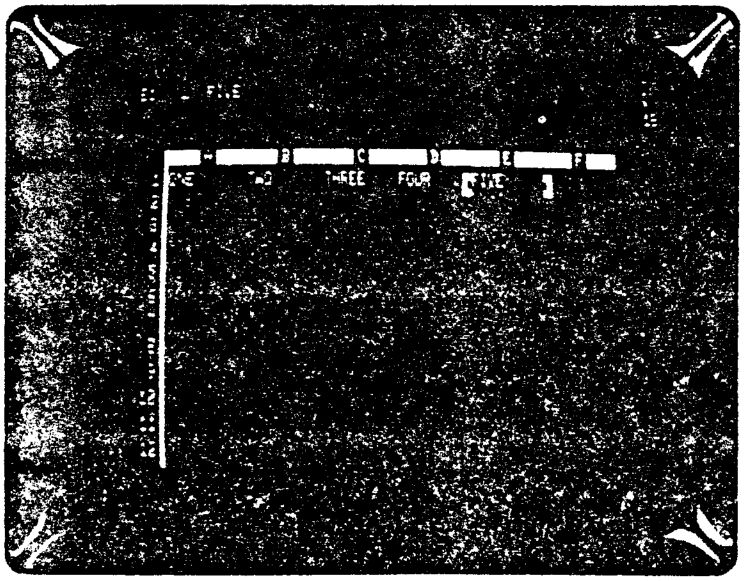
2) Type /M	prompt line: edit line:	Move: From . . . To E1
3) Press .	edit line:	E1 . . .
4) Press ◆◆	edit line:	E1 . . . C1
5) Press Ⓔ	Column E has been moved to Column C.	



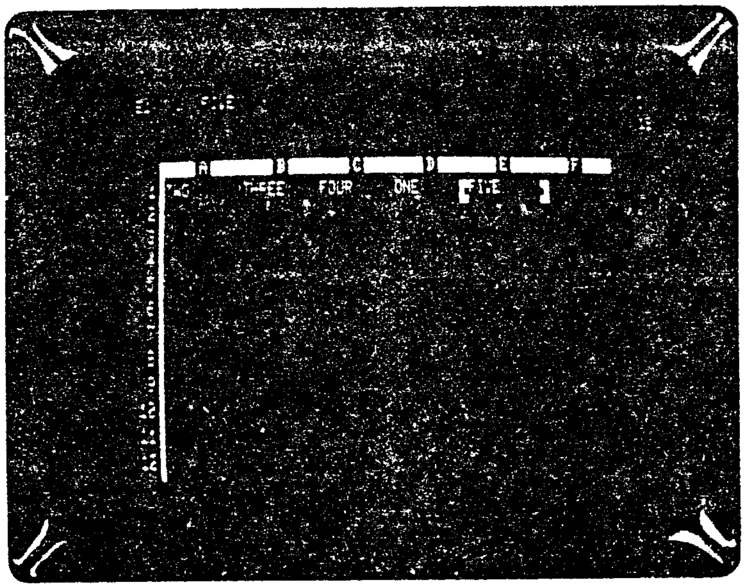
It is also possible to type in the coordinate of the row or column to be moved as well as the coordinate of the new location. Use the CLEAR key to erase any unwanted coordinate and replace it, either by typing or pointing to the coordinate with the cursor.

Example

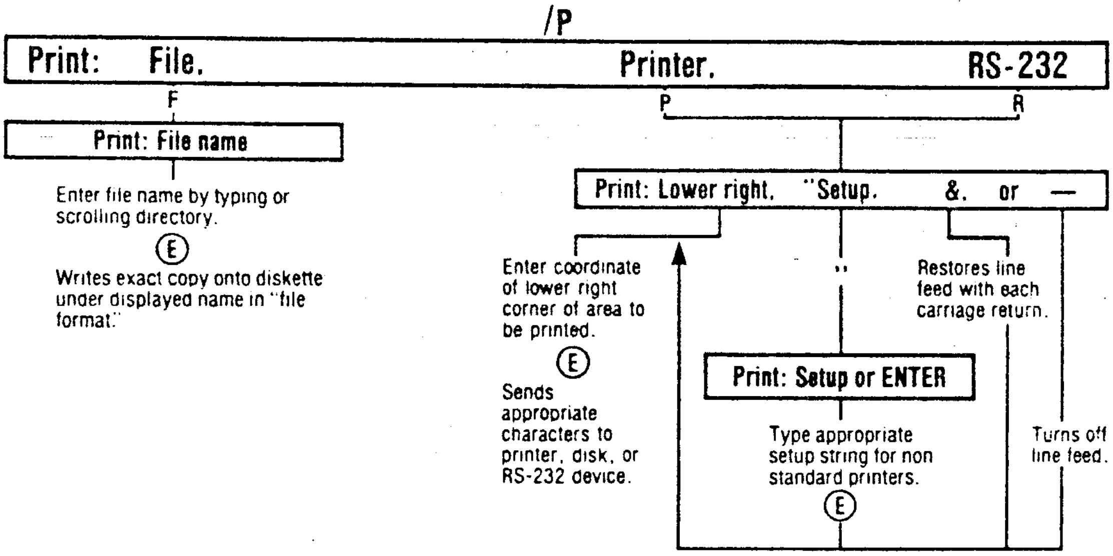
1) Set up your sheet as in the photo below, with the highlight on E1:



2) Type /M	prompt line: edit line:	Move: From . . . To E1
3) Press CLEAR	edit line:	blank
4) Press ◀◀◀◀	edit line: Column A will be moved.	A1
5) Press .	edit line:	A1 . . .
6) Type E1	edit line: Column A will be moved to column D. Although the cursor is on E1 at the end of step 5, pressing Ⓔ at that point will not enter E1 on the edit line.	A1 . . . E1
7) Now press Ⓔ	The contents of column A are now in D. The columns that were previously B, C, and D have been moved to A, B, and C.	



The PRINT Command


/P
Print: File. Printer. RS-232
F
P
R
Print: File name
Enter file name by typing or scrolling directory.
E
Writes exact copy onto diskette under displayed name in "file format."
Print: Lower right. "Setup. &. or —
Enter coordinate of lower right corner of area to be printed.
E
Sends appropriate characters to printer, disk, or RS-232 device.
"
Restores line feed with each carriage return.
Print: Setup or ENTER
Type appropriate setup string for non standard printers.
E
Turns off line feed.

The print command sends a copy of all or part of a VisiCalc sheet to a printer. This manual gives instructions for printing on a TRS-80 Line Printer, a "daisy wheel" printer, or a serial printer. A special form of the print command can be used to "print" a VisiCalc sheet to a disk file, to a modem or to other devices.

Before you begin printing, make sure your printer is properly connected, supplied with paper, and turned on. For instructions on setting up a TRS-80 Line Printer, see the Line Printer Operator's Manual (Catalog numbers 26-1150 or 26-1152). No two serial printers are quite alike; read your printer manual carefully.

You may need to use the TRSDOS FORMS command and SETCOM command before using your printer. If you have a parallel interface printer, you can set paper width and page length with the FORMS Command. If you have a serial interface printer, you need to use the SETCOM Command to activate the port and to set the parity, baud rate, etc. These commands are described in your TRS-80 Model III Owner's Manual.

The VisiCalc program prints rectangular portions of the electronic sheet. The position of the cursor when you type the **/P** command marks the upper left corner of a "printing rectangle." In giving the print command you will also specify the lower right corner of the rectangle. All rows and columns which lie within this rectangle will be printed. Window and title settings are ignored. If you try to print a row which has more characters than the line width of your printer, the extra characters in each row will either wrap around to a new line of printing, or will be ignored. Which of these two happens depends entirely on your particular printer. (See your printer specifications for this information.) The best way to print a sheet which is wider than what your printer can accommodate is to print it in separate sections and then tape the sections together to create a "hard copy" of the electronic sheet. To print the formulas and formats behind the data on your screen, see the Storage Command, /SS :P

To begin the printing process, use the arrow keys to move the cursor to the coordinate that marks the upper left corner of the rectangle you want to print.

1) Type **/P**	prompt line:	Print: File, Printer, RS-232

2) If you want to print to a TRS-80 Line Printer, type **P** (for Printer). If you have a serial printer, type **R** (for RS-232). The prompt line will display:

   Print: Lower right, "Setup, –, or &

This prompt means that you can now enter the coordinate of the lower right hand corner you want to print, or you can type one of the special setup characters (", **&**, or **–**). For the moment, ignore the setup characters. To enter the lower right coordinate, either type its position, or use the arrow keys to move the cursor to that coordinate. When you have done this, type Ⓔ and the printing will begin.

To interrupt printing at any time, press the BREAK key.

Before entering the lower right coordinate, you may optionally use any of the special setup characters which appear on the prompt line (in step 2, above). Typing a " results in the following prompt line

Setup or ENTER

Whatever characters you type next will be transmitted to the printer immediately. End the setup string by pressing Ⓔ

Many printers need the setup string to print in proper format. The setup string is nothing more than a special command to the printer. Setup strings usually begin with an ESCAPE or CONTROL character and are followed by another character or characters.

The VisiCalc program has a way to see these characters on the edit line. This method uses the caret (∧) in combination with other characters. The caret character is generated by pressing **SHIFT** and @ at the same time. The character(s) following the caret are treated specially. These characters will appear on the edit line, and you can check the setup string before you send it to the printer.

∧C	causes the character following to be treated as a control character.
∧E	causes the character following to be treated as an escape character.
∧**Hnn**	produces the single ASCII character defined by the hexadecimal nn.
∧**R**	puts a return character into the setup string.
∧L	puts a line feed into the setup string.
∧∧	puts one caret character into the setup string.

Each time you print you must reenter the setup string. When Ⓔ is pressed, the previous: Print: Lower right, "Setup, –, or & is again on the edit line and all three options are available. The setup string may not be printed at this time because some printers wait until receiving a carriage return character before printing.

The & and – characters are complementary. & forces a line feed between successive lines of printing, while – suppresses one. The & should be used if the rows overprint one another. The – should be used if your printer adds an extra line feed between lines of printing. If you do use either of these, there will not be any immediate effect, but you should see the results in the actual printing. Most printers will not require the use of & or –. For those that do, it may be possible to reset switches on the printer itself to permanently generate or suppress line feeds, so that you do not have to use the & or – characters. These may work differently on your output device. See the specifications for your device for more information.

The Print File Option

It is possible to "print" a VisiCalc sheet to a disk file, a modem, or other device. To write data to a file so that it can be retrieved both with a program of your own or by the VisiCalc program, use the Storage Command /S=S. The VisiCalc Storage Command /S=L will retrieve this data.

1) Type /**P** prompt line: Print: File, Printer, RS-232

2) Type **F** (for file) prompt line: Print: File name

3) Enter any legal file name here. File names are discussed in detail in the STORAGE Command. You may also use the directory scrolling feature. A file name may consist of up to 8 characters, and an optional extension of a / and up to three characters. Optional device names may be used as they are explained in the manuals for the use of TRSDOS. You specify the disk drive number with :0 or :1 or :2 or :3 appended to the file name. The standard TRSDOS file specifications may be used, including passwords. If you save a sheet using the /PF option, the VisiCalc program will automatically append /PRF to the file name. When using this option, do not use the /VC suffix, which identifies files saved with the /SS command (see the STORAGE Command).

The REPEATING LABEL Command

/ —

Label: Repeating

Type any characters.

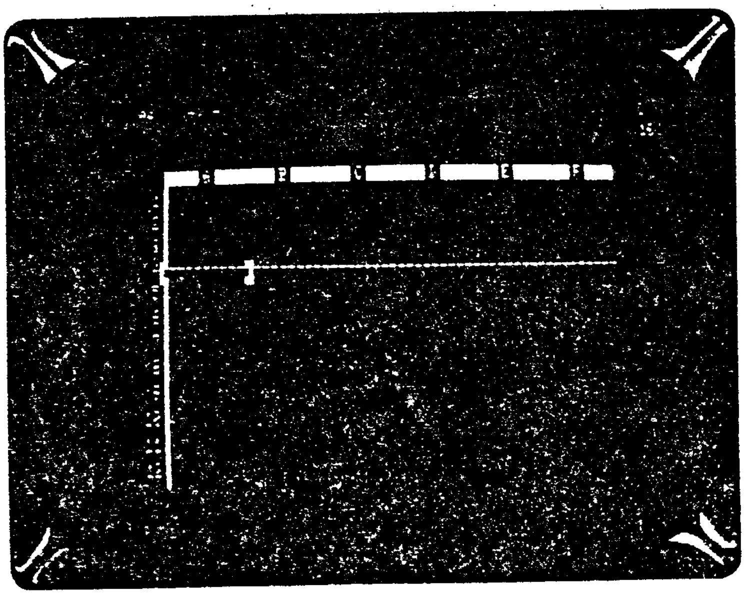
Ⓔ or ↑↓→←

Character sequence repeated across column width.

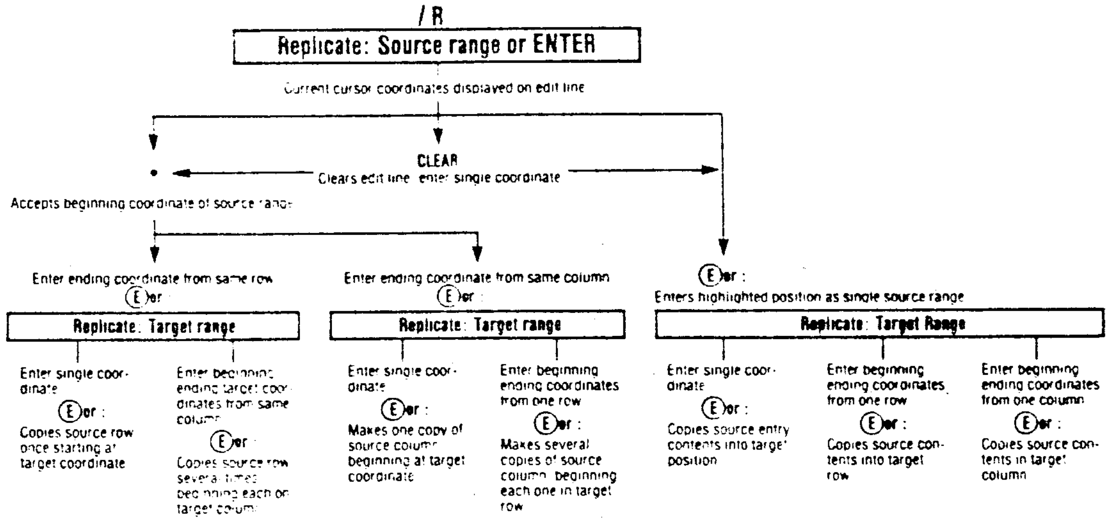
Sometimes it is useful to draw lines or other borders across an entire column or across several columns. The REPEATING LABEL command will repeat any sequence of characters you enter across the entire width of a column. If you change the column width (see the GLOBAL Command), the continuous sequence will be modified so that it still fills the column. When you replicate (see the REPLICATE Command) the entry containing the repeating label, you can form a line or other visual break across your sheet.

Example

1) Type >A4Ⓔ		
2) Type /—	prompt line:	Label: Repeating
3) Type —	prompt line: edit line:	Label: Repeating —
4) Press Ⓔ	The VisiCalc program will extend the sequence across the column. For example, with a general column width (9 characters) you see – – – – – – – – – in the entry position.	
5) Type /RⒺ B4.BK4Ⓔ	Use the replicate command to copy the continuous label as far across the sheet as needed.	



The REPLICATE Command


/R
Replicate: Source range or ENTER
Current cursor coordinates displayed on edit line
CLEAR
Clears edit line enter single coordinate
Accepts beginning coordinate of source range
Enter ending coordinate from same row
Ⓔor
Replicate: Target range
Enter single coordinate
Ⓔor :
Copies source row once starting at target coordinate
Enter beginning ending target coordinates from same column
Ⓔor
Copies source row several times beginning each on target column
Enter ending coordinate from same column
Ⓔor :
Replicate: Target range
Enter single coordinate
Ⓔor :
Makes one copy of source column beginning at target coordinate
Enter beginning ending coordinates from one row
Ⓔor :
Makes several copies of source column beginning each one in target row
Ⓔor :
Enters highlighted position as single source range
Replicate: Target Range
Enter single coordinate
Ⓔor :
Copies source entry contents into target position
Enter beginning ending coordinates from one row
Ⓔor :
Copies source contents into target row
Enter beginning ending coordinates from one column
Ⓔor :
Copies source contents in target column

The REPLICATE command allows you to copy the contents of any entry position, including labels, values (numbers, expressions, and formulas), formats (the FORMAT Command), and blank entries (the BLANK Command). You may copy a single entry from one place to another, copy a single entry position into a row or a column, or you may reproduce a single row or column as many times as you like. However, the REPLICATE command does not permit you to copy a row into a column, or to copy a matrix (several rows or columns at once). Complex copying operations must be done with a series of replications or with the /S=S option (see the STORAGE Command).

To use the REPLICATE command, you must supply two (and sometimes three) sets of information:

1. the source range

2. the target range

3. indication of the relationship of a value reference (see the VALUE ENTRY Command) to the new position

A range may be a single entry position, as well as a whole or partial row or column. Ranges are specified by typing or by pointing with the cursor to the desired beginning entry coordinate, then typing a period, then typing or pointing to the ending coordinate. A range may not cross the sheet diagonally. A complete source and target range specification will appear on the edit line in the following pattern:

beginning source entry position coordinate . . . ending source entry position coordinate:
first target entry position coordinate . . . ending target entry position coordinate

If the beginning and ending coordinates in your **source range** are identical, you will copy one entry only. If they are different but fall in the same column, you will copy that section of that column. If they are different and fall in a row, you will copy that section of that row.

The coordinates you put in the target range tell the VisiCalc program the starting position for each copy to be made with the REPLICATE command. If you want one copy of an entry position, a row, or a column, your target range should contain one coordinate (F9, for example). If you specify two different coordinates for your target range (F9 . . . F15, for example), you will get multiple copies of your source.

Copying from One Entry Position to Another

Give one coordinate as the source range and one as the target range.

Example

1) Type >**A1**Ⓔ	This places the cursor over the entry position you want to copy. If A1 is blank, type **100**Ⓔ
2) Type /R	prompt line: Replicate: Source range or ENTER edit line: A1 By starting the command with the cursor on A1, the VisiCalc program automatically entered A1 to begin the source range.

3) Press Ⓔ	prompt line: Replicate: Target range edit line: A1 . . . A1: You have told the VisiCalc program to begin copying the contents from entry position A1 and to end with A1. A1 is the "source range" consisting of one entry position.
4) Type **D1**	prompt line: Replicate: Target range edit line: A1 . . . A1: D1 This identifies entry position D1 as the start of the "target range," that is, the entry position to which the contents of A1 will be copied.
5) Press Ⓔ	The replication is completed. The value 100 is now in entry position D1 as well as A1, the highlight is still at A1, and the control panel has been cleared for a new command.

Creating a Column by Making Several Copies of One Entry

Give one coordinate as the source range and two coordinates within a single column as the target range:

Example

Repeat steps 1 through 4 in the example above. Change step 5 to:

5) Type **.D100**Ⓔ	The contents of entry position A1 will now appear in entry positions D2, D3, D4, through D100, as well as in D1.

This procedure is especially useful for setting up display formats (see the FORMAT Command) before entering a large group of numbers. Assume, for instance, that a column will contain sales figures and therefore should always display numbers rounded to two decimal places. Place the cursor on a blank entry position and type **/F$** (see the FORMAT Command) to attach the "dollars and cents" format to the position. Then replicate that entry position into the positions in the column that you want to have the dollars and cents format, using the procedure in the example above. The entry positions will not change in appearance if there are no values in them at this point. However, when you begin entering numbers into them, they will all be displayed with two decimal places.

Copying a Column from One Position on the Sheet to Another

Give the top and bottom entry position coordinates of the column as the source range: For instance, A1 . . . A32. For the target range, give only the coordinate of the top entry position of the new column.

Example

To set up your sheet for this example, enter the numbers 1 through 10 in positions A1 through A10. You need not clear the sheet because the VisiCalc program will write over any old entries.

1) Type **/R**	prompt line: Replicate: Source range or ENTER edit line: current cursor position coordinate. The VisiCalc program will put the coordinate of the entry position to which the cursor is pointing on the edit line. If this is not the coordinate you want to begin your source range, press **CLEAR** to remove it from the edit line.
2) Press **CLEAR**	To erase first coordinate (this step is not necessary when the coordinate on the edit line is the one you want to begin the source range).
3) Type **A1**	edit line: A1
4) Type **.**	edit line: A1 . . .
5) Type **A10**	edit line: A1 . . . A10
6) Press Ⓔ	prompt line: Replicate: Target range edit line: A1 . . . A10:
7) Type **C4**	edit line: A1 . . . A10: C4
8) Press Ⓔ	The contents from your source range (A1 to A10) are now in your target (C4 to C13). Note that you must not type the ending coordinate of the target range. The cursor has been returned to the position it occupied before you typed /R.

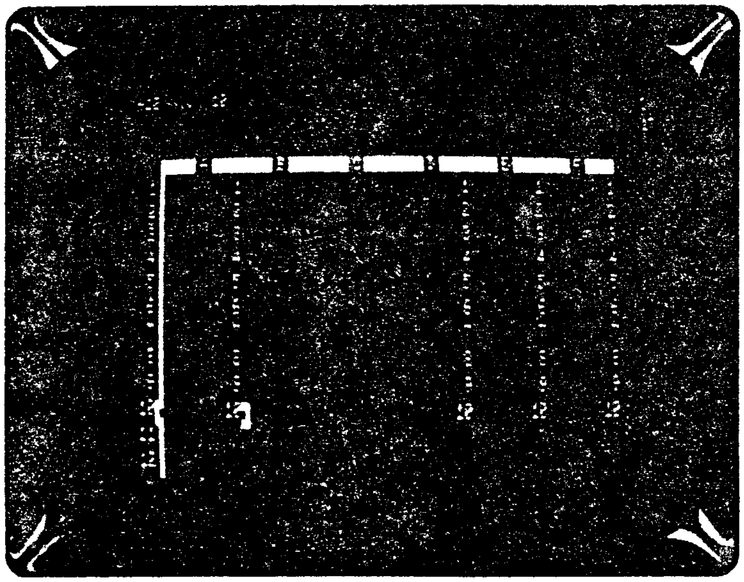
Making Several Copies of a Column

Enter the top and bottom coordinates of the column as the source range and the beginning and ending coordinates of a row as the target range. Each copy of the source column will "begin" in the target row.

Example

Do the example above again, only this time change steps 7 and 8 to:

7) Type **D1.K1**	edit line: A1 . . . A10:D1 . . . K1 In the target range, D1 will be the top of the first new column, E1 the top of the second new column, and so on, ending with K1 as the top of the last new column. The target range must be adjacent coordinates in one row or the VisiCalc program will only copy the column once and stop.
8) Type Ⓔ	The results will resemble this:



You may wish to scroll your window over to K1 to see all of the effects of these few keystrokes on your sheet.

Copying a Row from One Position to Another

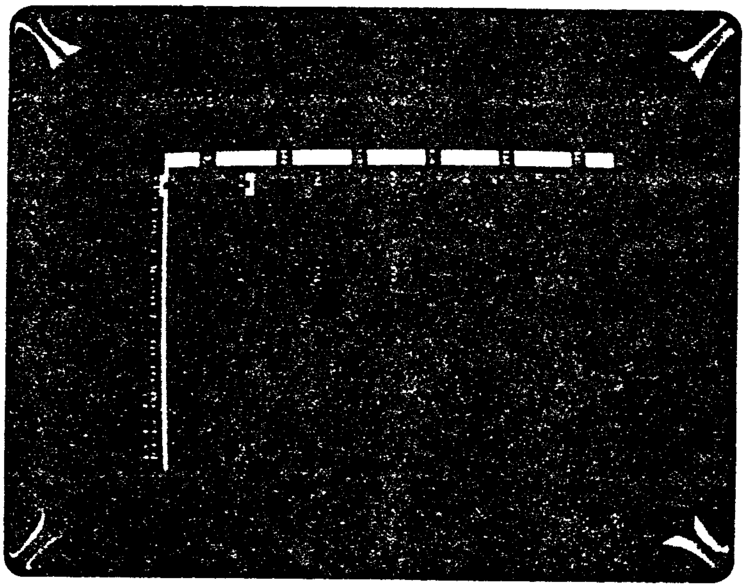
Specify the beginning and ending coordinates in that row as your source range. Then give the beginning coordinate only for the row in which you want the copy to appear. The VisiCalc program will automatically interpret this target coordinate as the first entry position in a row and will fill in the correct ending position.

Example

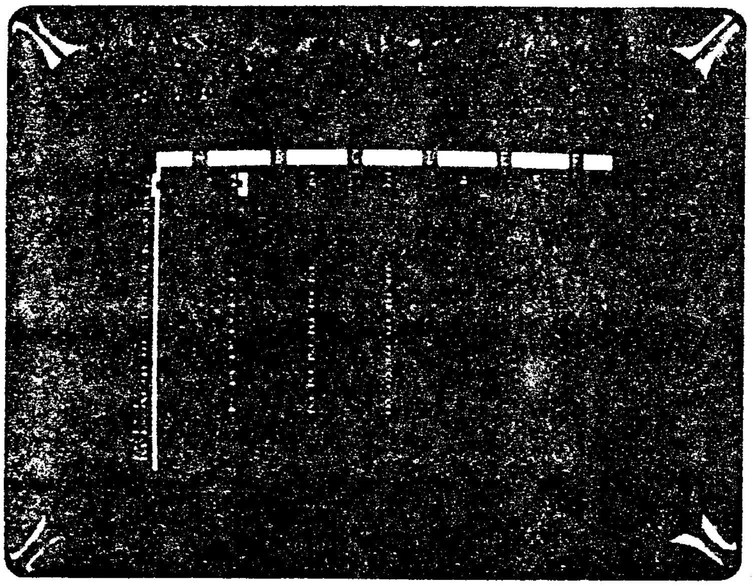
To set up for this example, clear the sheet with /CY and enter the numbers 1 through 5 across the top row on your sheet (positions A1, B1, C1, D1, E1). Then:

1) Type >**A1**Ⓔ
2) Type /**R.C1**Ⓔ edit line: A1 . . . C1:
3) Type **A5**
4) Press Ⓔ

The result should resemble this:

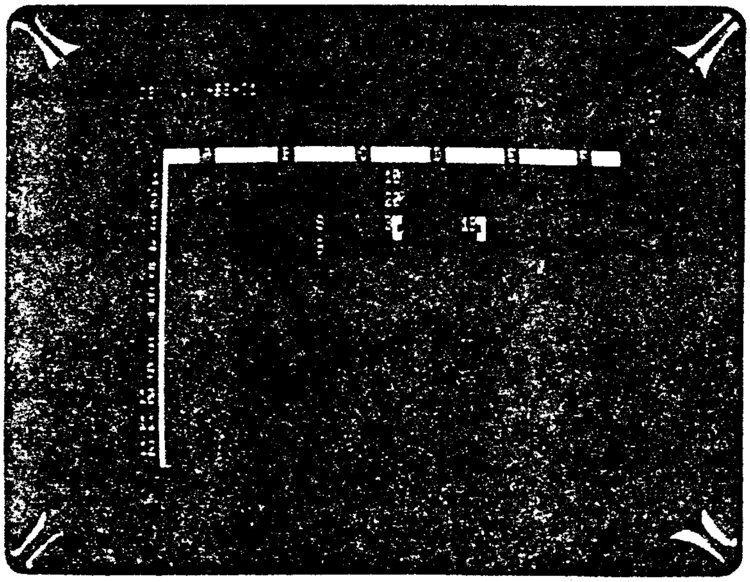


Give two coordinates for the target range (A5 . . . A10, for example) only if you want these results:



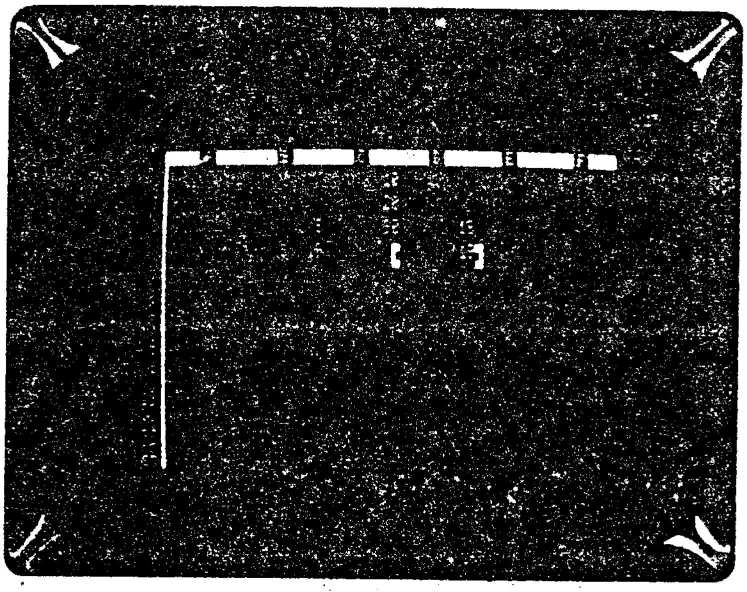
Replicating Value References

When you replicate a formula which contains entry position coordinates (value references), you must tell the VisiCalc program whether or not to copy each value reference exactly as it appears in the source range, or to change that value reference as it is copied. The change will replace the original value reference with the one that falls in the position that is relative to the location of the copied formula. Therefore, the replicated value reference will be to the replicated formula as the original value reference is to the original formula. The following examples will illustrate this relationship:



You can see from the entry contents line in the photo above that the formula in position D3 contains value references to positions B3 and C1. B3 is in the same row as the highlighted formula and two columns left. C1 is two rows up and one column over from D3.

Look at the photo below.



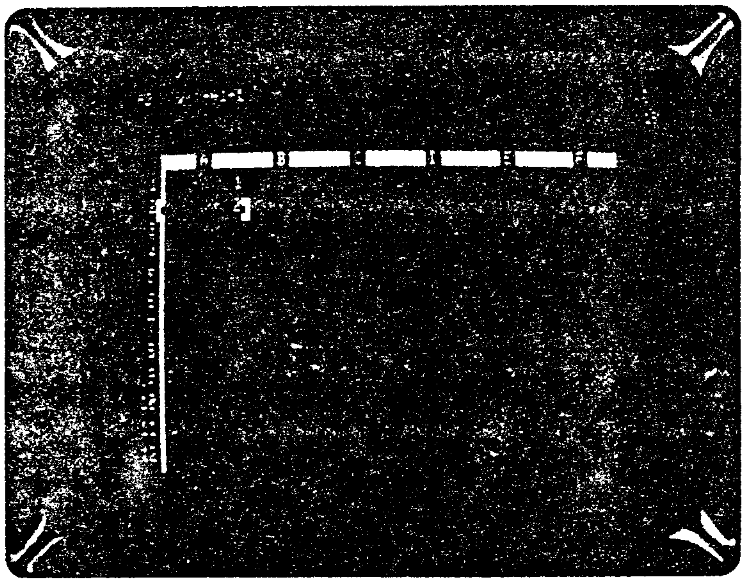
The formula in D3 has been replicated into position D4 but it has also been changed. The formula in D4 is now B4 + C1. The value reference B4 is in the same row and two columns left of the new formula in D4, i.e., it is in the same **relative position**. It was copied using the R indication. The value reference C1 is exactly the same in the formula at D3 and the formula at D4. In other words, it has not been changed to maintain the same position relative to the placement of the formula. It was copied using the N indication.

When a source range that is being replicated contains value references, the VisiCalc program places each value reference on the edit line with a highlight on it. The prompt line instructs you to type **R** if the value reference is to be relative, or **N** if it is not to be changed in the new formula. After this procedure has been completed for each value reference, the VisiCalc program will finish the replication.

Example

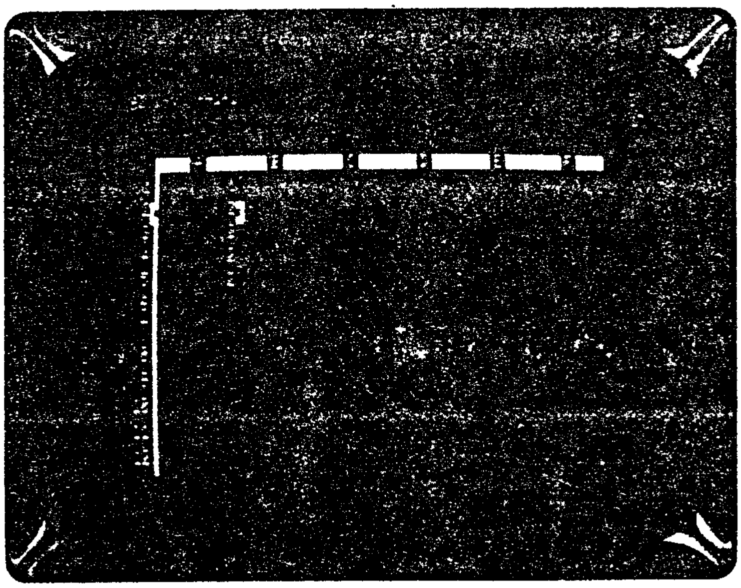
1) Type /CY — To clear the sheet.

2) Set up your sheet as in this photo (Notice the entry contents line for A2):

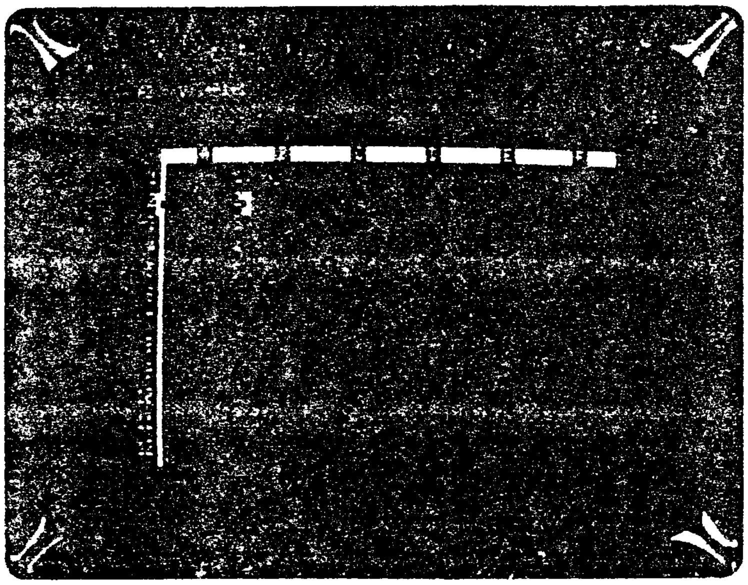


3) Type /RⒺ	prompt line: edit line:	Replicate: Target range A2 . . . A2:
4) Press →.	prompt line: edit line:	Replicate: Target range A2 . . . A2:A3 . . .

5) Press ◆◆◆Ⓔ	prompt line: Replicate: N=No Change. R=Relative edit line: A2:A3 . . . A5: +A1 The value reference A1 in the formula being replicated is highlighted. Replication actually takes place one entry position at a time. The edit line indicates that the first operation is copying from A2 into A3 and the VisiCalc program is waiting to be told whether to interpret the value reference A1 as relative or unchanging.
6) Type **N**	Your sheet will resemble this:



Move the cursor down the column. Notice that each copy of the formula contains the value reference, A1. It has not been changed from the way it appeared in the source range. Now repeat steps 1 through 5 in the example above. At step 6, type R Your sheet should resemble this:

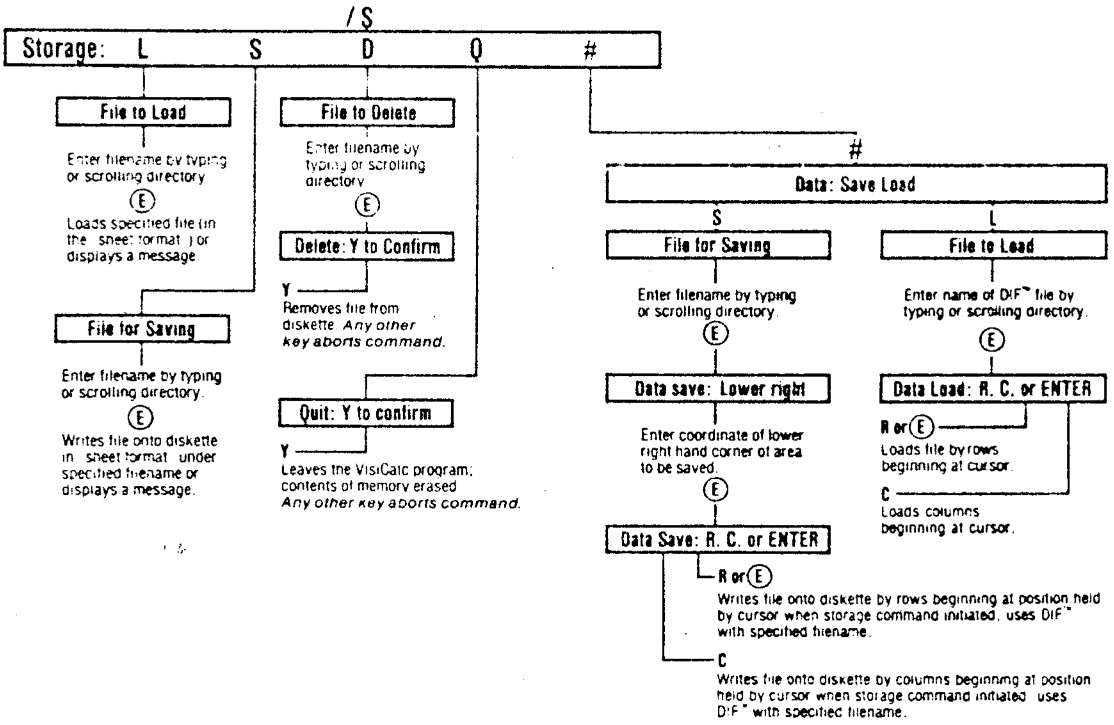


Move the cursor over A3, A4, and A5 again and check the formulas as shown on the entry contents line. The value reference in each entry position has been changed to cite "the position above" the highlighted formula (a relative description) instead of "the value reference which occurred in the source formula."

It is possible to replicate a formula into a position in which proper relative value references cannot be assigned. Turn back to the two screen photos at the beginning of this section ("Replicating Value References"). If, for example, you copy the formula at D3 into position A3, then the relative position of the value reference B3 will be located off the sheet to the left!

In Part II, Lesson Three of this manual, there are more examples of the use of the REPLICATE feature. Also, you must be careful not to introduce forward or circular references or incorrect calculation order into your sheet when you replicate formulas. These topics are thoroughly discussed in Part II, Lesson Three, in the sections entitled "The Order of Recalculation" and "Forward and Circular References" and also in Part III, the VALUE ENTRY Command.

The STORAGE Command



Enter filename by typing or scrolling directory. Ⓔ Loads specified file (in the sheet format) or displays a message.

Enter filename by typing or scrolling directory. Ⓔ Writes file onto diskette in sheet format under specified filename or displays a message.

Enter filename by typing or scrolling directory. Ⓔ

Y Removes file from diskette. *Any other key aborts command.*

Y Leaves the VisiCalc program; contents of memory erased. *Any other key aborts command.*

Enter filename by typing or scrolling directory. Ⓔ

Enter coordinate of lower right hand corner of area to be saved. Ⓔ

R or Ⓔ Writes file onto diskette by rows beginning at position held by cursor when storage command initiated, uses DIF™ with specified filename.

C Writes file onto diskette by columns beginning at position held by cursor when storage command initiated, uses DIF™ with specified filename.

Enter name of DIF™ file by typing or scrolling directory. Ⓔ

R or Ⓔ Loads file by rows beginning at cursor.

C Loads columns beginning at cursor.

The STORAGE command lets you save copies of your current VisiCalc sheet on diskette, load saved sheets back into the computer, and gives you a way to quit the VisiCalc program. One STORAGE command allows you to "store" or print the VisiCalc formulas to the printer. Because most STORAGE commands depend on proper use of file names, we will begin with a discussion of file names.

File Names

Any electronic sheet that has been saved on diskette is a file. You must assign each file a name unique to the diskette it occupies. When you save any file to diskette, the file name and the address of where the file is located on the diskette are saved in that diskette's directory. If you were to use a file name that you had already used, the more recent file would be written over the older file, erasing it. File names, however, are unique only to an individual diskette. Files saved to other devices than a disk drive, such as a printer, do not require a file name.

A valid file name can be no longer than eight characters in length, the first of which must be a letter. The rest of the name can include numbers. Thus, FILETWO and FILE2 are both valid. File names cannot include punctuation, the space character or any control or other special characters. FILE TWO or FILE,TWO are invalid and, if entered, will give an error message. A file name in upper case letters is unique from that same file name in lower case. Thus FILETWO is distinct from filetwo or FILEtwo.

File Name Suffixes

Besides the file name proper, the VisiCalc program also recognizes suffixes. Suffixes are used to identify a VisiCalc file and a device such as a disk drive to which a file can be written or from which it can be read.

The VisiCalc program automatically adds a suffix on to any file name when you save its file. This suffix allows the VisiCalc program to distinguish among the different types of files that it and other programs generate. It is added after the file name and before the drive number. Most common to the VisiCalc program is the /VC suffix, indicating that the file is a VisiCalc electronic sheet. The VisiCalc program will add a /VC suffix to any file that is saved by the VisiCalc program. You do not have to use the /VC in your file name, but you may. Thus FILE2 and FILE2/VC are equivalent.

To generate a print format file, type /PF and give the file name. Press ENTER and give the Lower Right coordinate. Press ENTER again, and the VisiCalc program will save your file with a /PRF suffix. PRF files are text files composed of ASCII strings. As such they are accessible by other programs.

Finally, if a file has been generated under DIF™, a /DIF suffix will be appended. Neither the /VC, the /PRF, or the /DIF suffix will count against the eight characters allowed for valid file name.

You may override the suffixes the VisiCalc program adds by creating your own, such as /ABC or /XYZ. The VisiCalc program will only add the /VC, /DIF, or /PRF suffix if you don't add a suffix. However, merely adding the suffix does not generate the file in the desired form. For example, if you give the command to write a file in DIF™ format, but add the suffix /VC yourself to the file name, the file will be in DIF™ format. An attempt by the VisiCalc program to load it back into your computer either as a /VC file or a /DIF file will give an error. The file is therefore unusable to the VisiCalc program.

The :n suffix is necessary only if you use two or more disk drives. If, after loading the VisiCalc program from drive 0, you wish to access another drive you must tell the VisiCalc program with the appropriate suffix by typing a colon and the drive number after the file name. Thus, for a file with the file name FILE2 in drive 1, you would type FILE2:1 If you use the /VC, the name might be FILE2/VC:1

Once a drive other than 0 has been used as a suffix, it becomes the "default" drive. That is, the VisiCalc program automatically looks to that drive whenever any drive command is given. To return the default condition to drive 0, you must add the suffix :0 after a file name though nothing keeps you from making drive 2 or drive 3 your next default drive.

The suffix :P can be used to save your screen's formulas and formats as a report on the printer. After typing **/SS** you have the prompt: Storage: File for Saving. At this point, instead of typing a file name, merely type a **:P** and press ENTER. The printer will print out both the formulas behind the values on the screen and the formats in effect. By printing out your sheet with the /P command (giving a printout of data only) and then printing it out with /SS :P to get a hard copy of the formulas, you will have a complete printed record of your sheet. Besides being a backup of your files, this is helpful in locating circular references.

To allow you to find and load or save your files more quickly, the VisiCalc program allows you to scroll through file names from diskette. To scroll through file names, type **/SL** then press the ↓. The drive will run and then display the first file name it finds in the directory on the edit line. If that isn't the file you want, press ↓ again. You can repeat this procedure until you have scrolled through every file name on the disk. To select a file for loading, simply press Ⓔ when the name of the file you want appears. You can scroll through both /VC and /DIF files.

You can also use scrolling to modify a file name already in the directory—a handy technique for updating. For example, if you had loaded and modified a file called SALES3 and wanted the update version to be called SALES4, you would type /SS↓ Continue to press the ↓ until the name SALES3/VC appears on the edit line. Then erase the 3 by using **CLEAR** and replace it with 4. Because the VisiCalc program will reappend the /VC suffix, just press Ⓔ to save the current sheet as SALES4.

For a complete discussion of file names, see the section on "File Specification" in your *TRS-80 Model III Owner's Manual.*

The Storage Load Command—/SL

This command loads back into the computer's memory a sheet which was saved with the /SS command, as discussed below. Only files which were saved in the VisiCalc file format with the /SS command can be loaded with the /SL command.

During file loading, temporary ERROR messages may appear. When the file has loaded, the sheet will appear in the window exactly as it was when you gave the /SS command to save it. If ERROR messages remain, your sheet may contain forward or circular references. The VisiCalc program will not clear the sheet that is in use when the /SL command is given, but will write the loaded sheet over it. Wherever an entry position has contents on both the current and the loaded sheet, the entry position contents of the just-loaded sheet will replace the previous contents. Blank entry positions in the loaded sheet will not erase the contents of corresponding entry positions in the old sheet. This over-writing characteristic gives you one way to combine sheets by loading previously saved sheets over one another.

Clear the sheet by typing **/CY** (see the CLEAR Command) before loading a sheet when you want only the contents of the saved sheet to be placed in the computer's memory and on the screen.

Insert the diskette containing the file to be loaded and follow the steps below. You may put the diskette into any of your disk drives and the VisiCalc program will search through the file directories on each one until the requested file is found.

1) Type **/S**	prompt line:	Storage: L S D Q #
2) Type **L**	prompt line:	File to Load
3) Type the file name used when the file was saved with /SS or Press ➧		The file name of the first file that was saved in the VisiCalc file format with the /SS command will appear. Each time you press the ➧ key, the VisiCalc program will examine the directory and place the next qualifying file name it finds on the edit line. Continue pressing ➧ to **scroll** through the directory until the name of the file you wish to load appears on the edit line.
4) Press Ⓔ		The VisiCalc program will load the file with the name that was on the edit line. While the file is loading, characters will flash at the left side of the edit line. When loading is finished, the sheet will appear in the window, looking just like it did when you saved it.

The directory **scrolling** described in step 3, above, puts the existing file names on the edit line. When a file name is on the edit line, you may change the name by backing up the edit cursor with CLEAR and then typing the characters you wish. This feature might be useful when, for example, you have forgotten a file name. As you scroll the directory a file name appears that jogs your memory as to the name of the file you want to load. Perhaps the name on the edit line is BUDGJAN/VC and you remember that the file you want is called BUDGAUG/VC. Press the CLEAR key until JAN/VC has been deleted from the edit line, type AUG/VC, then press Ⓔ and BUDGAUG/VC will be loaded (if it is on a diskette that is currently in one of the disk drives).

The Storage Save Command—/SS

This command will save an electronic sheet, just as you have created it, on a diskette in the VisiCalc format so that it can be loaded back into the computer's memory with the /SL command. Before you can use a diskette with the STORAGE SAVE command, a diskette must be initialized with the TRSDOS BACKUP commands as explained in the Introduction to this manual. All formulas, labels, title settings, explicit and global formats and window settings as well as the position of the cursor will be saved and in force when the sheet is loaded.

Each sheet is saved, or recorded, on a diskette under a name you specify, called the file name. The file name is recorded in the diskette's "directory." Each file name is unique, so that if you save a sheet with a file name already in the directory, the file with that file name will be replaced by the most recent file saved with the name. There is enough room on a diskette to hold many electronic sheets. Should the disk become full while the VisiCalc program is recording a sheet into a file, an ERROR message will appear on the edit line. The VisiCalc program will have saved all that it could under the file name you gave so you will want to delete that incomplete file from the full disk (see /SD below) and then save the sheet on another, less full, storage diskette.

The file name may be made up of 1 to 8 alphanumeric characters beginning with a letter. Then add the extension /VC (for the VisiCalc file format). File names are part of a more complete description of your file called a file specification (file spec. for short). Please refer to your *TRS-80 Model III Owner's Manual*, section entitled TRSDOS, subsection File Specification, for a full explanation of TRSDOS file specifications. All you need to give the VisiCalc program is the file name when saving a sheet. Following the file name you give a : and the number of the disk drive. You may type all or part of the complete file specification whenever you name a file. However, do use caution if you use a password in your file spec. (1 to 8 alphanumeric characters preceded by a (period)). Passwords are used to keep other people from tampering with your files. The password part of the file specification never shows up in the directory. To load such a file, you must type filename.password onto the edit line. If you forget your password you are in deep trouble!

When you load the VisiCalc program at the beginning of a session, your working copy of the VisiCalc program diskette must be in drive 0. Unless you specify a drive number (with :number after the filename), the computer will automatically assume that the diskette on which you want to save a file is located in drive 0. Drive 0 is said to be the "default" drive If you want to save the file on a diskette in drive 1, you must follow the file name you enter with the characters :1. For example, MYFILE/VC:1 would be saved on the diskette in drive 1. The :1 is the "drive specification." You can use :P as a device specification for the printer. This prints the formulas behind the values on the printer. Type **/SS** When you see the prompt File for Saving, type **:P**

To save a file, do the steps listed below. The cursor may be anywhere on the sheet when the /SS command is started.

1) Type **/S** prompt line: Storage: L S D Q =
2) Type **S** prompt line: File for Saving
3) Type the file name and the disk drive number, if necessary. or you can press ▶ key and the VisiCalc program will place the first file name it finds in the directory on the edit line. If there is no file name in the diskette's directory, the VisiCalc program will clear the control panel and wait for another command. **Note that your sheet has not been saved, because there is no file name.** Pressing Ⓔ will not save your sheet. (In this case, start at step 1 and type in a valid file name.) Each time ▶ is pressed, the next file name in the directory will appear, as the VisiCalc program scrolls through the directory. You may change the file name by pressing **CLEAR** to back up the edit cursor and then type in the new characters. When you have the desired name do step 4. Be careful not to press ◀ by mistake because this will respond like Ⓔ.
4) Press Ⓔ The disk drive will run as the electronic sheet is saved on the diskette under the file name you specified on the edit line in step 3.

The option, in step 3, of letting the VisiCalc program put file names from the directory on the edit line can be very useful. For instance, if you wanted to keep a record of different versions of a VisiCalc sheet, you might identify the versions by changing only the last few characters of each file name. You could scroll the directory until the last version's file name appeared, then backspace with **CLEAR** to delete the version identifier, then type in the new version characters. For example, using this method, the file name BUDGETA/VC:1 could readily be edited to be BUDGETB/VC:1. It is also a convenient way to examine what file names are in the diskette directory so that you do not unintentionally use an existing file name for the file being saved. Only files with /VC extensions will be displayed. Saving a file with a file name that is already in the directory will cause the old file to be written over. The VisiCalc program requires confirmation before it will write over a file. You will see the prompt File exists: Type Y to Replace. If you type Y the new file will be written over the old one. If you press any other key you will be returned to the command level.

The Storage Delete Command—/SD

This command will delete from the diskette the file whose name appears on the edit line. This command can also be used to scroll through all files on a diskette regardless of suffix, and regardless of whether a file is actually deleted.

1)	Type **/SD**	prompt line:	File to Delete
2)	Type the file name, followed by the drive number, if necessary. Or you may press ▶ and the VisiCalc program will place the first name it finds in the diskette's directory on the edit line. Each time you press ▶, the VisiCalc program will examine the directory and place the next /VC file name it finds on the edit line. Continue pressing ▶ to scroll through the directory until the name of the file you wish to delete appears on the edit line.		
3)	Press Ⓔ	prompt line: Any other key will cancel the command.	Delete: Y to confirm
4)	Type **Y**	The VisiCalc program will delete the file from the diskette. This command will only work on diskettes with the write protect notch uncovered.	

The Storage Quit Command—/SQ

This command will let you leave the VisiCalc program and return to the TRSDOS READY prompt (the same one that's on the screen when you first turn on the computer). **Any VisiCalc sheet currently in the computer's memory will be lost.** You will have to reload the VisiCalc program (see step 2 below) if you wish to continue using it. You should have a diskette with TRSDOS on it in Drive 0.

1)	Type **/SQ**	prompt line:	Quit: Type Y to confirm
2)	Type **Y**	The computer leaves the VisiCalc program. Any other key cancels the command. To begin the VisiCalc program again, with a VisiCalc diskette in drive 0, type **VC** Ⓔ	

Saving Files in DIF™ Format

Files saved with the command /S#S are recorded on the diskette in DIF™ format. This format affords a way for other programs, such as those written in BASIC or FORTRAN, to use the data that is on the sheet. The data that was saved in files with the /S#S command can be loaded back onto a VisiCalc sheet with the /S#L (Load a DIF™ file) command. You do not need to understand what the data interchange format is in order to use it.

Saving a DIF File—/S#S

This command saves a rectangular area of the sheet, which you define, in a file in DIF™ format. This command saves labels, blanks, and calculated values as they appear in the entry positions on the sheet. The formulas from which the values were derived are not saved (see the VALUE ENTRY Command for discussion of formulas and values). The DIF™ format allows for two orientations of the data. You can specify to the VisiCalc program which orientation you want by pressing **R** (row) or **C** (column) at the appropriate time. For data that you are only going to use with VisiCalc, always save the data with the R specification. As an aid to remembering this, pressing Ⓔ is the same as typing **R** To use the command:

1) Position the cursor at the entry position that is in the upper left hand corner of the rectangular area you want to save.		
2) Type /S#	prompt line:	Data: Save Load
3) Type S	prompt line:	Data save: File for Saving
4) Type the file name	The file name must follow the same rules outlined in the discussion of /SS, above, except that the VisiCalc program will append the suffix /DIF to files saved with this command. Do not use the /VC suffix, which is for use with files saved with /SS.	
Or press ⬍	To look at the DIF file names already in the directory. Continue pressing ⬍ until the desired file name appears on the edit line. You may change the name on the edit line by press **CLEAR** and typing other characters.	
5) Press Ⓔ	prompt line:	Data save: Lower right
6) Type or point with the cursor to the entry position in the lower right corner of the rectangular area to be saved. You may save only one column or one row if you want.		
7) Press Ⓔ	prompt line:	Data save: R C or ENTER
8) Press **R** or Ⓔ	The rectangular area of the sheet defined by the upper left and lower right cursor positions will be saved.	

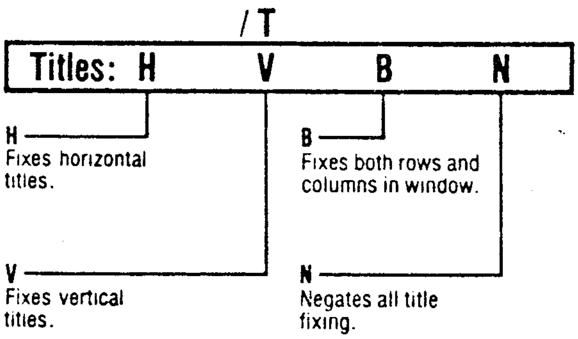
Loading a DIF File—/S#L

This command will load back the data you saved with the /S#S command into any position on the sheet you indicate.

1) Position the cursor on the entry position in the upper left corner of the area to be filled by the data being loaded.		
2) Type **/S#L**	prompt line:	Data Load: File to Load
3) Type the desired file name	The file name should be the one saved in DIF™ format.	
Or press ▶	To scroll through the directory as described in the /SL command discussion, above, until the desired file name appears.	
4) Press Ⓔ	prompt line:	Data Load: R C or ENTER
5) Press Ⓔ or **R**	The selected file is read into the computer's memory and placed on the VisiCalc sheet starting at the current cursor position.	

If you would like the loaded data transposed—what would have been loaded across the row is loaded down the columns, and vice versa—press **C** at step 5.

The TITLES Command



Most VisiCalc sheets are considerably larger than the screen display window. To see all the entries on the sheet, you must move the window away from the top and left edges, so that any row and column titles you may have entered move out of sight.

The titles commands allow you to fix titles in place on the screen, so that they remain in view as you scroll the window about the sheet. Begin the command from a cleared control panel by typing /T The prompt line on the control panel will display Titles: H V B N. The possible keystrokes are:

H	To fix **horizontal** titles.
V	To fix **vertical** titles.
B	To fix **both** horizontal and vertical titles.
N	**No** titles (to "unfix" titles).

Which columns and/or rows are to be fixed is determined by the position of the cursor when you initiate the command. All rows at and above the highlight are fixed by H. All columns at and to the left of the highlight are fixed by V. B fixes all rows above and columns to the left of the highlight.

The VisiCalc program terminates the command automatically and clears the control panel immediately. There is no change on the sheet, but the effect of title fixing becomes apparent when you begin scrolling the window away from the top and left borders.

You cannot use the arrow keys to move the highlight to an entry position that is within a fixed title area. The highlight cursor and the coordinate in the upper left hand corner will flash when it bumps into the fixed titles. You jump the title barrier by using > (see the GO TO Command) and the coordinates of an entry position within the fixed title area; for example, **>A1Ⓔ**

In certain circumstances, the VisiCalc program will automatically unfix vertical titles you have set. It does this when in order to follow your commands, the vertical titles must not be set. For instance, if you scrolled the window so that column B was at the left edge, then fixed vertical titles and then at some point gave the command to go to a coordinate in column A (see the GO TO Command), the VisiCalc program would obey the command to go to column A, scrolling to the left to bring A into view. If you have fixed vertical titles, and then you expand the column width (see the GLOBAL Command) so that only one column can be displayed on the screen at one time, the VisiCalc program will display column A. When you return to a narrower column width, you will have to refix the vertical titles. The TITLES command also allows you to unfix titles.

All values are stored with between 11 and 12 significant digits. When a number is displayed in the general format (see the FORMAT Command), the VisiCalc program will shift between conventional and scientific notation as required to display the calculated value with the greatest precision. In scientific notation, the number 123456789123 becomes 1.235E11. The "E11" means "times 10 to the 11th power." The largest number possible is .999999999999E62. The smallest is 9.999999999E−66. Except in $ format (see the FORMAT Command), non-significant zeros are dropped from the display.

If the column is too narrow to display a number, even in scientific notation, the VisiCalc program will place as many right angle brackets (>) as it can in the entry position.

An illegal calculation, such as dividing by zero, will result in a value that is displayed as ERROR in the position where the illegal calculation occurs and in all other positions that reference the calculation.

Any of the following keystrokes will initiate the value entry command: **a digit** **+ − (. # @**

The arithmetic operators which separate elements in formulas are:

+	**addition**
−	**subtraction**
/	**division**
*	**multiplication**
^	**exponentiation**

The VisiCalc program performs the calculations in the order it encounters them, from left to right. To change the order of precedence, use parentheses to indicate "do this first." If there are parentheses within parentheses, the VisiCalc program will calculate the innermost first. For example, 5+6/2*4 will be evaluated as 22, but 5+((6/2)*4) will evaluate as 17.

The complexity of a formula, i.e., the number of value references, arithmetic operators, parentheses, functions and their arguments, and the amount of memory in your computer determine the maximum length of the formula you may enter. If the formula becomes too complex as you type it, the VisiCalc program will stop displaying your keystrokes. You will be able to enter everything in the formula up to that point. The VisiCalc program will not allow you to enter an illegal formula such as one which ends with an arithmetic operator. You must back up with the **CLEAR** key to erase the offending character(s).

A formula that does not contain value references may be one such as 1435 or − 14.35 or 5*12+1−60. As soon as you type one of the characters that initiate a value entry (a digit + − (. # @) the VisiCalc program immediately places Value on the prompt line and displays the character you typed on the edit line. As you continue to type in the elements of the formula, they appear on the edit line. Until you have terminated the command (and entered the formula) by pressing Ⓔ or one of the cursor-moving keys, you can use **CLEAR** to modify what you've typed, or press BREAK to interrupt the value entry command and clear the control panel.

When the value entry has been made, the VisiCalc program displays the **calculated** value in the entry position on the sheet, but stores in its memory the actual formula that was on the edit line when you terminated the command. The formula that was on the edit line when you finished the value entry will always be displayed on the entry contents line when you place the cursor on the value.

Example

1) Type **/CY**	This clears the sheet. Highlight is on A1.	
2) Type **5**	entry contents line:	A1
	prompt line:	Value
	edit line:	5
3) Type ***12+1−60**	edit line:	5*12+1−60
	position A1:	blank
4) Press Ⓔ	entry contents line:	A1 (V) 5*12+1−60
	prompt line:	blank
	edit line:	blank
	position A1:	1

About Value References

The VisiCalc program permits entry position coordinates, such as D10, to be used as elements in formulas. These elements, called **value references**, take on the value in the entry position to which they refer. This value will change whenever you change the contents of the referenced entry position. You cannot just begin typing in a formula that begins with a value reference, since the first character you would type would be a letter and the VisiCalc program would assume you are typing a label (see the LABEL ENTRY Command). You must initiate the value entry command with a + or a − and then type or point with the cursor to the value reference. If the next keystroke is an arithmetic operator, you may follow it with a number or a value reference. Until you've terminated the value entry command with Ⓔ, ←, →, ↑, or ↓, you can use **CLEAR** to back up and change the formula on the edit line, or press BREAK to interrupt the command and clear the control panel.

The cursor-moving keys (↑, ↓, ←, →) will not terminate the value entry command if they were just used to point to a value reference that follows an arithmetic operator on the edit line. For example, if 1+ is on the edit line, and you point with the cursor to A5, the formula on the edit line will be 1+A5. Pressing a cursor-moving key at this point will not terminate the command, but will change the coordinate A5 to correspond to the new cursor position. In this case, Ⓔ must be used to enter the formula.

Example

1) Type **/CY**	This clears the sheet. Highlight is on A1.	
2) Type **1→1→**	position A1:	1
	position B1:	1
	Highlight is on C1.	
3) Type **A1**	prompt line:	Label
	edit line:	A1
	Typing the letter A has started the label entry command.	
4) Press **BREAK**	To clear the control panel. Highlight should still be on C1.	
5) Type **+**	prompt line:	Value
	edit line:	+
6) Type **A1+←**	edit line:	+A1+B1
7) Press Ⓔ	entry contents line:	C1 (V) +A1+B1
	prompt line:	blank
	edit line:	blank
	position C1:	2

Every time you change the value at A1 or B1 in the example above, you will see position C1 change accordingly.

If you are looking at different parts of the sheet through a split screen (see the WINDOW Command), you can use the ; to jump from one window to another to point to entry position coordinates you wish to use in the formula.

Use of ! and

The VisiCalc program may also be used like a calculator as you are typing in a formula on the edit line. Suppose you wanted to store the result of 5*12+1−60 as the value entry, and not the formula from which it was derived. Simply type the formula on the edit line and then press the exclamation point (!). The VisiCalc program immediately calculates the value. If Ⓔ or one of the cursor-moving keys is pressed at this point, the number will appear in the entry position and in the entry contents line. Alternatively, you could continue developing the formula for the value entry.

If a formula contains any value references, the VisiCalc program will get the current value in the entry position referenced and calculate the formula for you on the edit line.

Example

1) Type **/CY**	This clears the sheet. Highlight is on A1.	
2) Type **4*5*7Ⓔ**	entry contents line: position A1:	A1 (V) 4*5*7 140
3) Type **▶1+◀**	edit line:	1+A1
4) Press **!**	edit line:	141
5) Press Ⓔ	entry contents line: position B1:	B1 (V) 141 141

Pressing # will place the numerical value of an entry position on the edit line. If you press # immediately after a value reference on the edit line, the value of the coordinate will replace the value reference on the edit line.

If you start the value entry command by pressing #, the VisiCalc program will place the value of the highlighted entry position on the edit line. A value of 0 will appear if the highlighted entry position is blank or contains a label (see the LABEL ENTRY Command) or if you've entered a value of 0.

Only the **calculated** value in the entry position will be placed on the edit line, and not the formula from which it was derived.

There are several ways you might want to use this feature: to place the value of a highlighted entry position on the edit line when you start the value entry command; to duplicate a value in one entry position into another; to place the value in an entry position onto the edit line when you are developing a formula.

Example

1) Type **/CY**	This clears the sheet. Highlight is on A1.
2) Type **10↓10↓10Ⓔ**	prompt line: blank edit line: blank Position A1, A2, and A3 each contain the value entries of 10. Highlight is on A3.
3) Press #	prompt line: Value edit line: 10
4) Type **/2Ⓔ**	entry contents line: A3 (V) 10/2 position A3: 5
5) Type →→	entry contents line: C3 position C3: blank Note that the highlight has been placed on the position into which we want to make an entry.
6) Press +	prompt line: Value edit line: +
7) Press ←←	edit line: +A3
8) Press #	edit line: +5
9) Press Ⓔ	entry contents line: C3 (V) 5 position C3: 5 The cursor has returned to C3, the position at which the command was started when you typed + at step 6. Had you pressed # before +, 0 would have appeared on the edit line since a blank entry position has a value of 0.
10) Press ↓	Cursor is at C4, which is blank.
11) Type # + ↑ + ←←↑	edit line: 0+C3+A3
12) Press #	edit line: 0+C3+5 Note that # returned the value only for the value reference immediately preceding it.
13) Type ***A1#**	edit line: 0+C3+5*10 You may type in a coordinate as well as point to it.
14) Press Ⓔ	entry contents line: 0+C3+5*10 position C4: 100

Example

This example illustrates the use of both ! and #:

1) Type /CY	To clear the sheet. Cursor is at A1.	
2) Type **1◄**	position A1: Cursor is on A2.	1
3) Type **+**	prompt line: edit line:	Value +
4) Press ▲	edit line:	+A1
5) Type **+1**	edit line:	+A1+1
6) Press Ⓔ	entry contents line: position A2:	A2 (V) +A1+1 2
7) Type **◄1+▲#**	edit line: The value of A2 has replaced the value reference.	1+2
8) Type **!**	edit line: The formula on the edit line has been evaluated.	3
9) Press Ⓔ	entry contents line: position A3: Notice that the formula used to derive the value of 3 has not been stored because of the use of !.	A3 (V) 3 3

In this example, every time you change the value of A1, you will see the effect of that change in position A2, which contains A1 as a value reference in the formula +A1+1. Try it by placing the cursor on A1 and typing **100**Ⓔ The value in A2 will be recalculated and replaced with the new value, 101.

Recalculation Order

VisiCalc formulas may contain as many value references as the complexity of the formula will allow. When any value entry is made, including changing an existing entry, the VisiCalc program automatically recalculates every value on the sheet. Recalculation always starts in the upper left hand entry position, A1. When first loaded, the VisiCalc program calculates a value for A1, then A2, then A3, then A4 to the end of column A. Then it calculates B1, B2, B3, B4 to the end of column B; then C1, C2, C3, C4 and so on. Note the letter C in the upper right corner of the control panel. This indicates that the order of recalculation for the whole sheet is by Column. The global command (see the GLOBAL Command) contains an option which lets you change the order of calculation from down columns to across rows. When row calculation is in effect the upper right corner of the control panel will display an R.

Forward and Circular References

Pay particular attention to the placement of any formulas which contain value references. When in column recalculation, be sure that all referenced entry positions are to the left of the formulas which cite them (or above a formula in the same column). If the sheet is not arranged in this way, the formula containing a value reference will be recalculated before the new value has been placed in the referenced entry position. When recalculation has been completed, the sheet will display the value of the formula as calculated using the old value from the referenced entry position. However, the new value of the referenced entry position will be displayed in the entry position.

This problem, called **forward referencing**, is often difficult to diagnose and one is tempted to conclude that the VisiCalc program has made an arithmetic error. If you suspect your sheet contains a forward reference which is causing a formula to be incorrectly updated, press the ! once. This will force another recalculation of the whole sheet. Watch the suspect formula. if a new value appears, look for forward references. @ERROR may be used to look for forward references. (See Function Arguments below.) You may choose to redesign your sheet to eliminate all forward references or to use multiple !'s for recalculation. In row calculation, referenced values must be placed in the rows above the formulas which use them or to the left in the same row.

A **circular reference** is one which cites itself. For example, placing the formula 1+A1 in entry position A1 is a circular reference. Each time the sheet is recalculated the value of this formula will change, even if no other changes are made on the sheet. DO NOT use circular or self references in a worksheet—their results are unpredictable. To calculate a running total, such as a month-to-date accumulation, study the @SUM function. A file loaded with circular references will show ERROR in the entry position.

Functions

Functions perform more complex calculation than simple addition, subtraction, multiplication, or division. Some functions save the effort of typing frequently used formulas (such as adding the values of a range of locations); some perform calculations that are not otherwise possible (such as trigonometric functions); and some choose alternative values for calculations (such as looking up tax rates from a table).

A function can be used anywhere a formula can be used. It consists of the name of the function (which always begins with @) followed by an *argument* (or list of arguments) in parentheses. An argument consists of the values (formulas and references to other locations) the function uses to calculate its own value.

The @SUM function, for example, could be written:

@SUM(B1,S2,A4*.23)

This adds the value of locations B1 and S2, and .23 times the value of location A4.

An example of the @SQRT function:

@SQRT(625)

Its value is 25, the square root of 625.

The @CHOOSE function selects one of several alternative values based on the value of the first argument:

@CHOOSE(A4,17,8,23,44)

The value of this function depends on the value of its first argument, A4. If A4 is 1, the value is 17; if A4 is 2, the value is 8; and so forth.

The @ starts a value entry (no preceding + is necessary).

Functions That Use A Single Argument

The arithmetic and trigonometric functions require a single argument.

Arithmetic Functions

The functions in the following table perform the listed arithmetic calculation on a single argument (specified by *v* in the table):

Function	Result
@ABS(*v*)	Absolute value of the argument
@EXP(*v*)	e (2.71828...) to the power specified by the argument.
@INT(*v*)	Integer portion of the argument.
@LN(*v*)	Natural log (base e) of the argument.
@LOG10(*v*)	Logarithm (base 10) of the argument.
@SQRT(*v*)	Square root of the argument.

Trigonometic Functions

The functions in the following table perform the listed trigonometric calculation on a single argument. All angle arguments or resultants must be specified in radians.

Function	Result
@SIN(*v*)	Sine of the argument.
@COS(*v*)	Cosine of the argument.
@TAN(*v*)	Tangent of the argument.
@ASIN(*v*)	Arc sine of the argument.
@ACOS(*v*)	Arc cosine of the argument.
@ATAN(*v*)	Arc tangent of the argument.

Functions That Use a List of Arguments

The functions in the following table perform a calculation with a list of arguments (represented by *list* in the table). The arguments are separated with commas:

Function	Result
@AVERAGE(*list*)	Arithmetic mean of the values in the list. The result is equivalent to @SUM(*list*) divided by @COUNT(*list*).
@COUNT(*list*)	Number of non-blank entries in the list.
@MAX(*list*)	Largest value in the list.
@MIN(*list*)	Smallest value in the list.
@SUM(*list*)	Sum of each value in the list.

Net Present Value—@NPV

The @NPV function calculates the net present value of future cash flows. It takes two arguments: the discount rate, or cost of money, used to discount the future cash flows, and a range of locations that include the cash flows themselves.

Functions Without Arguments

Several functions do not require an argument.

@PI

@PI is the ratio of the circumference of a circle to its diameter, 3.1415926536.

@NA and @ERROR

@NA (Not Available) is used when a worksheet is set up before the data is written. Because a blank location evaluates to 0, ERROR is displayed at each location where zero appears as a denominator and incorrect or misleading values can be produced at other locations.

Writing @NA at the blank locations causes the VisiCalc program to evaluate all entries that refer to those positions as NA. All formulas on the worksheet are legal.

The result of an illegal calculation is displayed on the worksheet as **ERROR**. This can be caused by a too-deep nesting of (or +, an error in writing a formula, or deleting a row or column that is referenced in a formula at another location. ERROR is displayed at the entry position that contains the error and all locations that refer to it.

The @ERROR function causes **ERROR** to be displayed at the location where it is entered and all locations that refer to it.

@TRUE and @FALSE

@TRUE and @FALSE are used with the logical functions described later in this section. They cause **TRUE** or **FALSE** to be displayed at the locations where they are entered. The entries **TRUE** and **FALSE** are also displayed when the comparison operators (<, >, =, <=, >=, and < >) are used.

Logical Functions

A logical value is one whose value is either TRUE or FALSE. A logical function is one that manipulates logical values. Logical calculations are similar to mathematical operations, but operate only on this special set of values.

Comparison Operators

The comparison operators work on two numeric values and evaluate to a logical value. For example, the formula 4 > 1—four is greater than one—evaluates to the logical value @TRUE. The formula 5=3—five is equal to three—evaluates to the logical value @FALSE.

These comparisons can be used either as entries on the worksheet or arguments in a logical function. If used as entries on the worksheet, they should not be written at a location referenced by a function that does not use logical arguments. If this happens, the value of such functions is @ERROR.

The following table lists the comparison operators (it assumes the operator is preceded by *value1* and followed by *value2*):

Operator	Value
<	@TRUE if *value1* is less than *value2*, @FALSE if it is not.
>	@TRUE if *value1* is greater than *value2*, @FALSE if it is not.
=	@TRUE if *value1* is equal to *value2*, @FALSE if it is not.
<=	@TRUE if *value1* is less than or equal to *value2*, @FALSE if it is not.
>=	@TRUE if *value1* is greater than or equal to *value2*, @FALSE if it is not.
< >	@TRUE if *value1* is not equal to *value2*, @FALSE if it is.

@NOT

@NOT takes a single logical value as its argument (i.e., one whose value is @TRUE or @FALSE); the function's value is the opposite logical value. The value of @NOT(A1) is @FALSE if A1 is @TRUE and @TRUE if A1 is @FALSE. If the value of A1 is @NA, the value of @NOT(A1) is also @NA. if the value of the argument is anything other than @TRUE, @FALSE, or @NA, the value of the @NOT function is @ERROR.

@AND

@AND takes any number of arguments, each of which must be a logical value or a range of logical values. Its value is @TRUE if all the arguments are @TRUE, @FALSE if any of the arguments is @FALSE. If any of the arguments is not logical or evaluates to @ERROR, the value of @AND is @ERROR. If any of the arguments evaluates to @NA and all other arguments are @TRUE or @FALSE, the value of @AND is @NA.

@OR

@OR takes any number of arguments, each of which must be a logical value or a range of logical values. Its value is @TRUE if any of the values is @TRUE and @FALSE if all the values are @FALSE. If any of the arguments is not logical or evaluates to @ERROR, the value of @OR is @ERROR. If any of the arguments evaluates to @NA and all other values are @TRUE or @FALSE, the value of @OR is @NA.

@IF

@IF takes three arguments. The first must be a logical value; the second and third can be any value. The function evaluates to the value of the second or third argument, depending on the value of the first:

Value of First Argument	**Value of @IF**
@TRUE	Value of second argument
@FALSE	Value of third argument
@NA	@NA
Not logical or @ERROR	@ERROR

For example, the value of @IF(D5,2,3) is 2 if D5 is @TRUE and 3 if D5 is @FALSE. The value of @IF(D5,E1,E2) is the value of E1 if D5 is @TRUE and the value of E2 is D5 is @FALSE.

@ISNA and @ISERROR

@ISNA takes one argument. Its value is @TRUE if the value of the argument is @NA and @FALSE if the value of the argument is anything else.

@ISERROR takes one argument. Its value is @TRUE if the value of the argument is @ERROR, and @FALSE if the value of the argument is anything else.

@ISNA and @ISERROR are used to manipulate entries written as @NA or @ERROR and return a value that is not automatically designated @NA or @ERROR.

Functions That Select Alternative Values

Two functions can be used to select alternative values for calculations. These allow the worksheet to handle different situations or projections.

@CHOOSE

The @CHOOSE function takes one of the values in its list of arguments. The first element in the list is the index to the following arguments.

For example, in @CHOOSE(A4,17,6,33,39), A4 is evaluated first. If A4 is 1, the result is 17; if A4 is 2, the result is 6; and so on.

@LOOKUP

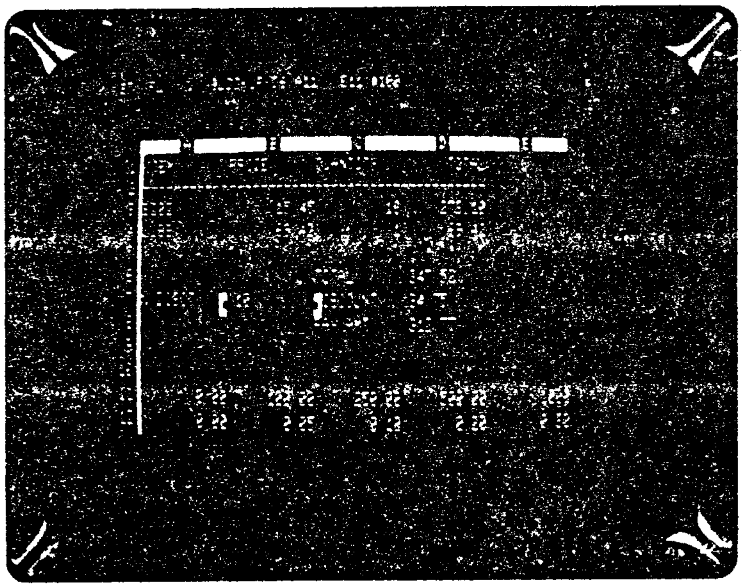
The @LOOKUP function looks up a value in a table, a form of calculation frequently used in financial calculations. Tax calculations, for example, require looking up gross pay in a tax table and using the corresponding tax rate.

@LOOKUP searches for a value in a range of locations and evaluates to a corresponding value from an adjacent range. The range to be searched can be in either a row or a column; the adjacent range must be in the column to the right of the range searched or the row below the range to be searched.

The value being looked up is compared to successive values in the range to be searched until a value is found that is larger than the value being looked up (or until the end of the table is reached). The entry in the adjacent range that precedes the match position is the value that the @LOOKUP function assumes.

Two arguments are required: the first is the value to be searched for, and the second specifies the range that contains the table of values to be searched.

The @LOOKUP function can be used to calculate an invoice. The total amount of goods purchased is looked up in a table, and the corresponding value from the adjacent range is used as the discount percentage to calculate the total amount of the invoice. Those calculations are shown in the following photograph:



The lookup table is in positions A11-E11. The cost of goods purchased is at D6. The @LOOKUP function is written at B7:

@LOOKUP(D6,A11...E11)*100

Multiplying the value by 100 makes it a percentage that is later divided by 100 (at D7). Although the lookup range is technically a forward reference, it makes no difference in this case because the values in the table are constants.

The formula at D7 is +D6*B7/100. The value at D11 is 500.00, corresponding to a discount of 0.20 at D12. If the invoiced items total 347.52, the discount is 10% (the value that correspond to 250.00). If the invoiced items total 3000.00, the discount is 30%, the same as for 1000.00.

The R in the upper right corner of the screen indicates that the worksheet is recalculated by row because the price-times-quantity calculations are made across rows.

Making the VisiCalc Program Less Precise

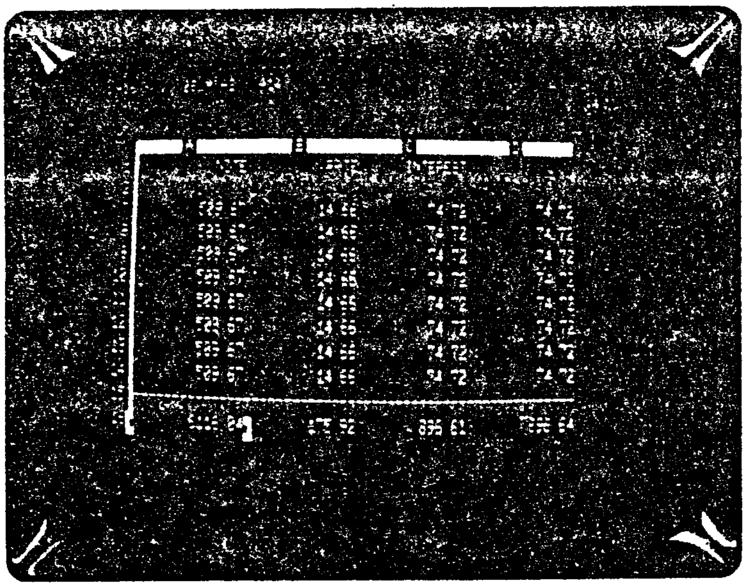
Because the VisiCalc program calculates and rounds to 11 (and sometimes 12) decimal places, differences of pennies and even dollars can occur between VisiCalc results and those produced by ordinary 2-place precision calculations. This difference can cause problems when attempting to balance books and during audits.

To solve this problem, you can use the following *rounding formula* to reduce the calculating precision to just two decimal places:

(@INT((*coord*)*100+.5))/100

coord is the location of the original formula.

For example, assume there are twelve notes worth $509.67 each. The interest paid on those notes is 14.66%. What will the value of the twelve notes be after one year? The following photograph illustrates the effect of the rounding formula:



The value 74.717622 is rounded to 74.72 *for display only*. Column C shows the result of multiplying the full precision by 12: 896.61. Column D shows the result of multiplying the rounded amount by 12 (the result produced by a calculator or paper and pencil): 894.64. If the difference of 3 cents isn't acceptable, a rounding formula can be used.

The formula at C3, which produces the more accurate result, is **+B3*B4**. The rounding formula, at D3 is:

(@INT((C3)*100+.5))/100

The rounding formula is replicated from D4 to D14; the @SUM function is used at D16 to add D3 through D14. This sum matches the less-precise calculation because the rounding formula has held precision to two decimal places.

The VERSION Command

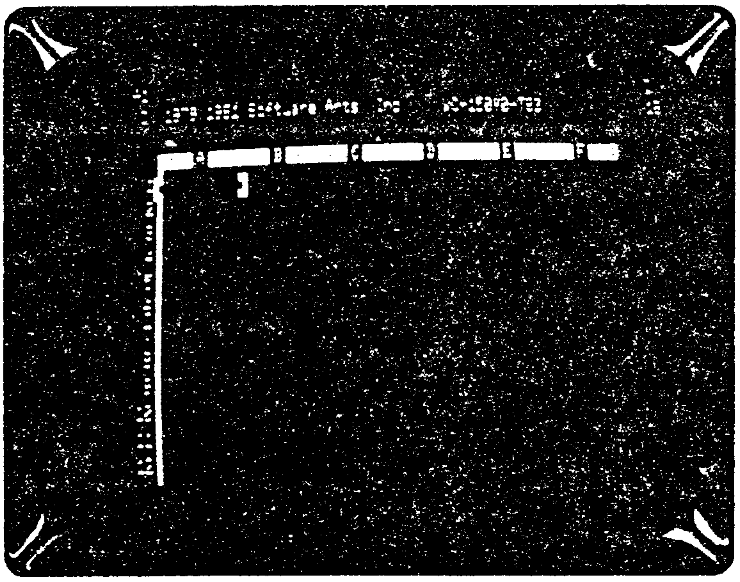
/V

```
Copyright ©1981
Software Arts Inc. x.xxx
```

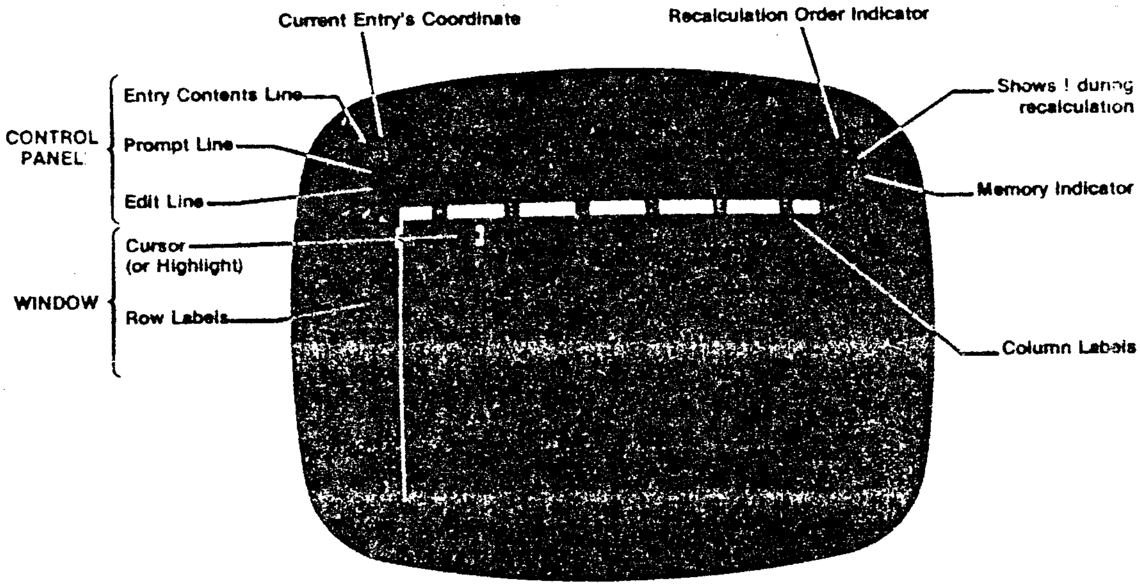
Displays copyright notice and version number on prompt line; clears automatically with next keystroke.

To see the copyright notice and the version number of your VisiCalc program, type /V when the control panel is clear. You may use this command anytime, without disturbing the contents of a sheet. As soon as you press any key, the notice will disappear from the control panel.

Should you need to call or write with questions about the VisiCalc program, be sure to include the version number that appears when you give the version command. The photograph illustrates how the notice appears on the screen.



The VisiCalc Screen



The Control Panel

The top three lines of the VisiCalc screen make up the **control panel.** Each line has a specific function:

The **entry contents line** is the top line of the panel. At the left, it will always show the coordinate on which the cursor is currently placed. Next to the cursor it displays the exact content (as it was entered) of that entry position, if there is anything written in it. Any explicit formats (see Part III, the FORMAT Command) that might have been set at the coordinate are also displayed here. At the right side of the entry contents line, the letter C or R appears, indicating that calculation and recalculation is being done down columns or across rows (see Part III, the GLOBAL Command).

The **prompt line** is the middle line in the control panel. At the left side, it displays the name of the command that is currently being used followed by a colon (:). Next comes a series of letters or words which indicate all possible choices offered by the VisiCalc program at this point. The number on the right indicates the amount of available memory left in the computer. On the command structure chart at the beginning of each command discussion in the VisiCalc Command Reference, each prompt line is enclosed in a box. Below the box are all the possible actions the VisiCalc program can take from that point in the command structure. The control panel is said to be "**clear**" when there is no prompt on this line. A command can be started only when this line is clear. (The memory indicator on the right will always be visible. However, the photo of the screen in this manual may have a different value for the memory indicator from what you see on your screen.) When a command is being used and there is a prompt on this line, pressing BREAK will cancel the command and clear the prompt line.

The **edit line** is the bottom line of the control panel. it displays each character you type or point to with the highlight cursor while using a command. An edit cursor always appears after the last valid character that was typed. Characters that are displayed on the edit line can be erased by backing up the edit cursor with the CLEAR key and then retyping, if desired. The VisiCalc program will also, on occasion, use this line to display information which it wants you to confirm or clarify before it carries out a command.

The Window and Sheet

Below the control panel is the VisiCalc **window** which looks upon a portion of the VisiCalc electronic sheet. Across the top of the window there is a border of letters, each of which is a column heading. The sheet is divided into 63 columns, labeled A, B, C, . . . BI, BJ, BK. Down the left side of the window there is a border of numbers, which serve as headings for each row. There are 254 possible rows on the VisiCalc sheet.

Entry Positions

The intersection of each column and row defines an **entry position.** A column letter and a row number identifies each entry position, *e.g.*, D17. This identifier is called the entry position **coordinate.**

The Cursor

When the VisiCalc program is loaded, there is a set of flashing brackets highlighting entry position A1. This bar is called the **cursor** and sometimes the "highlight." The cursor can be removed from the screen by typing **SHIFT CLEAR** and replaced by typing **SHIFT CLEAR** again. Typing **SHIFT 0** will make the cursor stop flashing; **SHIFT 0** again will start it flashing. Any command which performs an action on a single entry position will do so in the entry position that is highlighted by the cursor when the command is started. This coordinate remains displayed at the left side of the entry contents line until the command is completed and the cursor moved, although during the course of some commands the cursor can be moved (an action called **pointing**).

Moving the Cursor

The cursor can be moved to any position on the electronic sheet. The exact keys which cause the cursor to move are discussed in detail in Part I of this manual, in the section entitled "Some Notes on Your Keyboard." In the VisiCalc Command Reference, we continue to use the symbols →, ←, ↑, and ↓ to indicate the use of these keys to move the cursor. The cursor will move one entry position in the direction of the arrow each time you press the cursor-moving key.

When the cursor has been moved to the right or bottom edge of the sheet visible through the window, the VisiCalc program will **scroll** the entire window across or down the sheet, following the cursor so that it is in view at all times. To move around the sheet quickly, see Part III, the GO TO Command.

If the cursor is "bumping" into any of the four edges of the electronic sheet, the coordinate on the entry contents line will flash.

Automatic Repeat

Pressing ▼, ▲, ▶, or ◀ and **holding** it down will automatically cause the cursor to move more quickly, in the direction of the arrow. The window will scroll to keep up with the cursor.

Pointing with the Cursor

Whenever you are using a command and can type in a formula (see the VALUE ENTRY Command) or an entry position coordinate, the VisiCalc program allows you to move the cursor to **point** to the coordinate you want. Check the prompt and edit lines to be sure you have begun the command and the VisiCalc program is waiting for you to enter a coordinate before you press one of the cursor-moving keys. Then move the cursor to the desired entry position. You will see the coordinate on the edit line change as you move the cursor. If you try to point with the cursor when it is not allowed, the VisiCalc program may end the command and move the cursor to the next entry position.

Typeahead

At times, you may type faster than VisiCalc reacts to your keystrokes. This is because VisiCalc may be doing any number of things in reaction to the last key you pressed, such as expanding the electronic sheet and recalculating formulas. VisiCalc has a feature called **typeahead** so that it remembers the keys you pressed, no matter how fast you go. It will catch up with you as soon as it can. Use typeahead cautiously when entering commands and data together. It is easy to mix up keystrokes and enter a command as a label.

Correcting Mistakes

When you have characters on the edit line, you may back up to erase them by using the **CLEAR** key and then typing in the correct characters. Each time **CLEAR** is pressed, the edit cursor on the edit line will erase one character. Press it enough times and you'll back completely out of the command and have a clear control panel.

If you start a command and decide you don't want to complete it, press **BREAK** to cancel it and return to a clear control panel.

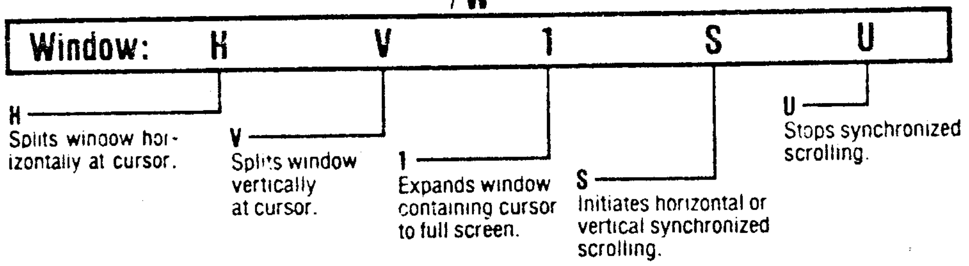
To change the contents of an entry position, move the cursor to the position of the coordinate, and initiate the desired command. When you finish the command with its appropriate terminator, the old contents will be replaced by the new.

The Memory Indicator

The number at the right side of the prompt line is called the **memory indicator** and tells you how much memory is available as you write on the electronic sheet. There is a complete discussion of the computer's memory and the way the VisiCalc program uses it in Part II, Lesson Two in the section entitled "Postscript: Memory and the Electronic Sheet."

The WINDOW Command

/W


Window: H V 1 S U
H
Splits window hor-izontally at cursor.
V
Splits window vertically at cursor.
1
Expands window containing cursor to full screen.
S
Initiates horizontal or vertical synchronized scrolling.
U
Stops synchronized scrolling.

Often you will find yourself wishing to compare rows or columns which are too far apart on the sheet to be displayed in a single window on your computer screen. The window command allows you to split your screen at the cursor position so that you can view the sheet through two windows on the sheet simultaneously. Each window may be independently scrolled around the entire sheet to let you see rows or columns which are widely separated on the sheet. You may also look at the same entry positions through separate windows with different global column widths and formats (see the GLOBAL Command) modifying the display in each one. Global column widths and formats are properties of the window or screen and do not change the internally stored data.

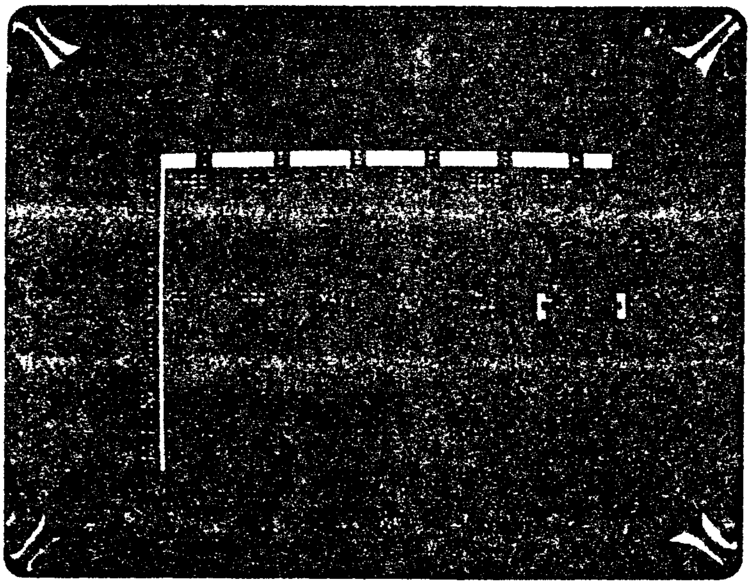
Horizontal Split—/WH

This command splits the window into two by placing a second column border (A, B, C, D,...) between the row containing the cursor and the next row down. Each window may be moved individually to view the same or different parts of the sheet.

Example

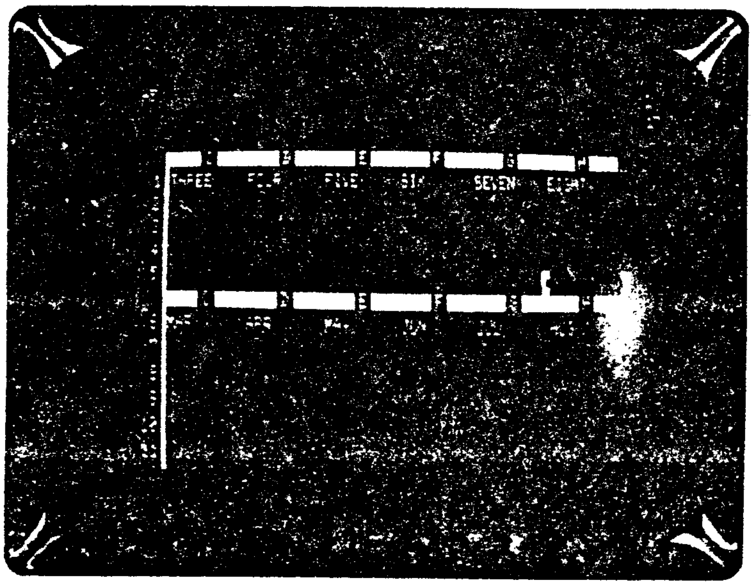
1) Type **/CY**	This clears the VisiCalc sheet.
2) Type **ONE◆TWO◆ THREE◆FOUR◆ FIVE◆SIX◆ SEVEN◆EIGHTⒺ**	
3) Type **>A6Ⓔ**	

4) Type
JAN♦FEB♦MAR♦
APR♦MAY♦JUN♦
JULY♦AUGⒺ
Your screen should resemble this:



5) Type **/W** — prompt line: Window: H V 1 S U

6) Type **H** — Your screen will change to:



Each window will now view the sheet independently. The cursor can move around the sheet in the top window. Move it up and left until the ONE is highlighted in the top window. Press ; to move the cursor into the bottom window. Pressing ; will always make the cursor jump to the last position it had in the other window. All VisiCalc commands will work in both windows and you can see the effects on the sheet through either window. The two exceptions are the /GC (Global Column) and /GF (Global Format) commands, which are set in one window at a time (see the GLOBAL Command).

Remove the horizontal window by typing **/W1** The window containing the cursor will then occupy the whole screen using the current format settings of that window. A horizontal window must be removed before a vertical window can be instated and vice versa.

Vertical Windows—/WV

This command splits the window by adding a second row border (1 2 3 4 5...) after the column containing the cursor. When the screen has been split vertically, all the columns in the right window may sometimes be one character narrower than those in the left window if that narrowing will preserve the column on the screen. If the column cannot be preserved, a column will be dropped from the vertically split screen. This is to make room for the second row border. The vertical window behaves exactly like the horizontal window described above. Note that when you return to one window, the format settings of the window in which the cursor is sitting go into effect in the single window. The column width of the righthand window will also be in effect if the cursor is in that window when /W1 is typed. To jump between windows, press ;. When the window is split vertically, you can reset the titles. The VisiCalc program will reset titles in the right window.

Example

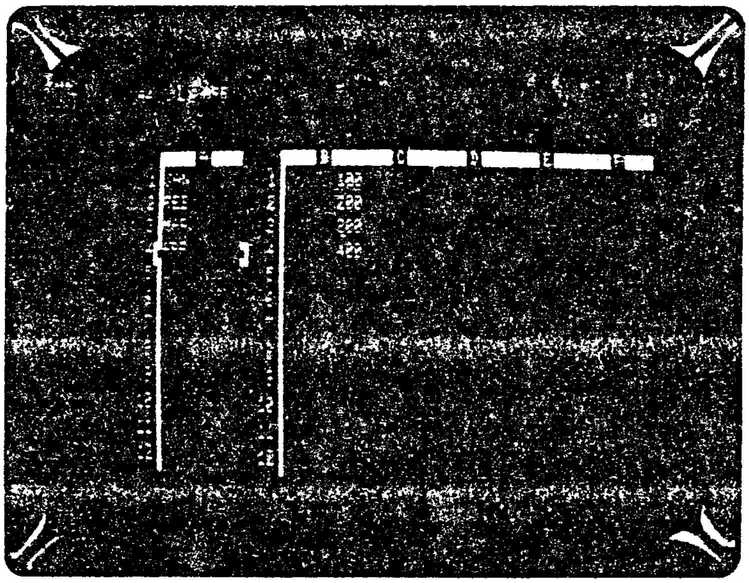
1) Type /CY

2) Type
JAN◆FEB◆
MAR◆APRⒺ

3) Type >B1Ⓔ

4) Type
100◆200◆
300◆400Ⓔ

5) Type /WV Your screen should resemble the photo below.



The vertical window can be particularly useful when you wish to keep a column of information visible in one window while you scroll the other window around to compare columns in different places on the sheet.

Return to One Window—/W1

This command displays the window containing the cursor in full screen position. All the global format settings in that window take effect in the one window.

Synchronized Scrolling—/WS

This command synchronizes horizontal motion in horizontal windows or vertical motion in vertical windows so that moving the highlight in one window also moves the other.

Unsynchronized Scrolling—/WU

This command turns off synchronized scrolling.

The last three window command options (/W1, /WS, and /WU) may only be used after a /WH or /WV is in effect.

PART IV. INDEX

-INDEX OF SPECIAL CHARACTERS-
(in ascending order by ASCII code)

Supplement. PROGRAMMER'S GUIDE to DIF™

Software Arts Technical Note SATN-18/TRS-80

Special Edition for the Radio Shack TRS-80 Model III

4. SAMPLE PROGRAMS

5. APPENDICES

Limited License to Copy

Rightful owners of this Guide are hereby licensed to copy it for their own use, provided that this notice is reproduced on each such copy. Copying for purposes of resale or license to others is prohibited.

No Warranty

This Guide is being published to enhance the usefulness of DIF™ used by the VisiCalc® program and other programs. NEITHER SOFTWARE ARTS, INC. NOR PERSONAL SOFTWARE INC. MAKES ANY WARRANTY, EXPRESS OR IMPLIED, WITH RESPECT TO THE QUALITY, ACCURACY OR FREEDOM FROM ERRORS OF THE DATA INTERCHANGE FORMAT OR OTHER CONTENTS OF THIS DOCUMENT, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY OR OF FITNESS FOR A PARTICULAR PURPOSE.

VisiCalc® a Trademark

The term "VisiCalc" is a registered trademark of Personal Software Inc. which designates a software product published by Personal Software Inc. under an exclusive license from Software Arts, Inc.

Software Arts, Inc.
(Reprinted by permission of Software Arts, Inc.)
10/80 V1.5II

1. Introduction

It is often desirable to process the same data by more than one program. For example, a data management system may be used to record sales values. These values are then to be used as the basis for projections using the VisiCalc program. Finally the projections may be plotted by a third program. How can you get data from one program to another without requiring the user to type the data in anew each time? Each of the programs processing the data may be written by a different person, and may even run on different machines.

In order to allow programs to "talk" to each other, we must agree upon a standard language. Software Arts, Inc., the creators of the VisiCalc program, have developed a data interchange format (DIF) that can be used as a common language for data. This is the format in which the VisiCalc program saves data with the /S# commands.

We are writing this document in order to explain to programmers how they can read and write data files using this format. The more programs that support the format, the more useful it becomes. The casual user should not be concerned about the details. It is only important to be aware that the format exists and that if two programs support the format, then it is likely that data produced by one can be processed by the other.

If you read this document fully, you will learn all of the details of the standard. This is not a tutorial, so you may find it helpful to skim the more technical parts that follow, and concentrate on the next section, the beginning of the Data File Format section, and the sample programs.

The sample programs in this document are all coded in TRS-80 BASIC.

2. What DIF Does and Does Not Do

The basic goal of DIF is to allow the interchange of data among a wide variety of programs. The type of data addressed by DIF is data that is stored in tables—columns and rows. Examples of this type of data would be time series, such as the daily closing price of one or more stocks that are to be input to a regression analysis package, or the actual expense figures for a company that are to be used as the starting point for a forecast. DIF treats all data as a group of equal length *vectors*—that is, groups of related data, like time series, or columns in a relation. The word vector is used, rather than column, since the actual orientation of the data (a horizontal row or vertical column) does not necessarily correspond to how it is logically oriented. Likewise, the corresponding elements of the vectors are called *tuples* rather than rows. For example, in the data below, the Sales, Cost and Profit figures (across the rows) could be viewed as vectors, with each year (down the columns) corresponding to a tuple:

YEAR	1980	1981	1982	1983
SALES	100	110	121	133
COST	80	88	97	106
PROFIT	20	22	24	27

The actual choice of which grouping of the data is considered to be the vectors, and which the tuples, is really up to the programmer or user. Some programs may just view the data as a rectangle of unrelated data, while others may require the user to be aware of the grouping. The VisiCalc program would be an example of the former, and a plotting package would be an example of the latter.

In DIF, data is stored by tuples. That is, it consists of successive values from each vector grouped together into tuples, which are then output (or input) in that order. In the data used for our example, if the vectors were across the rows (Sales would be one vector, Cost and Profit the other two), then the first tuple would consist of the three numbers 100, 80, and 20, in that order. The second tuple would be 110, 88, and 22, and so on.

When the VisiCalc program deals with data in DIF it gives you the option of storing or loading "by rows" (R or ENTER) or "by columns" (C). What the VisiCalc program means by "by rows" is that the vectors go across the rows, and the tuples go down the columns. For example, in our example data, saving Sales, Cost and Profit by rows would output first the tuple 100, 80, 20, and then the tuple 110, 88, 22, etc. "By columns" is just the opposite, with the vectors down the columns, and the tuples across the rows. For the same data, the first tuple by columns would be 100, 110, 121, 133, and then 80, 88, 97, 106, etc.

Not all of the programs that process the data stored in DIF will have identical requirements. For example, some programs will only be able to process a simple list of numbers while others will want to store attributes associated with multiple vectors of numbers. Thus, a goal in the design of DIF was that programs should be able to keep descriptive information about the data, but must not be *required* to generate it. At the same time, the program reading the data should be able to ignore all descriptive information that is not relevant to the actual processing of data.

The primary constraint on the format of data stored in DIF is simplicity. It should be very simple for users to write programs in a common language to read and write data files. Since BASIC is so pervasive and minimal, the needs of BASIC were used to determine the details of the format. It is necessary for other languages, such as Pascal or PL/I, to be able to process this data, too. Fortunately these languages allow the use of subroutine libraries. Thus, a standard set of subroutines to process the interchange format can be provided for the users of those languages, freeing them from many of the details of processing the data.

Nongoals were just as important as goals during the design of DIF. Specifically, there is no emphasis on a minimal space representation. This representation is meant to be modest and does not attempt to preserve the richness available in many database systems. The central idea is that we should be able to transport a table of values (numeric and/or string) from one program to another. There is additional mechanism to allow cooperating programs to exchange some information about the data, such as labeling.

Some of the more specific constraints are:

Predetermined data types
It is much simpler to write a program in BASIC if one knows ahead of time what the format of the data is, and in particular whether one is going to be reading a string or a number. Some BASICS are missing the VAL function that will convert from a string to a number, making it even more difficult. Therefore, DIF defines exactly which type of data is to be read at each point.

Lack of line input
Many BASICS do not have the ability to read a line of text without giving special meaning to some characters. For this reason strings containing special characters must be quoted.

Lack of parsing
Some BASICS will only input a whole line as a string. They do not use "," as a string value delimiter. Therefore, DIF always stores string values alone on a single line.

Input size
Many BASICS have a limited input buffer. 255 characters is a typical limit for the length of an input line. Therefore, DIF tries to keep most lines of information short.

Preallocation
In systems that permit dynamic allocation, it is often necessary to allocate the space before actually reading the data. Even when this is not required, knowing the total amount of data beforehand can be an important efficiency consideration. For this reason, DIF has a method for making this information available to a program reading the data.

End of data
In some systems it is either difficult or impossible to detect the end of data in a file gracefully. Thus the program should know when it has read the last value. DIF has a special provision to signal when the last data element has been read.

3. The Data File Format

A DIF file consists of two parts—the header and the data part. The header describes the data and the data part has the actual values. An example of a DIF file is the following, which is from our sample data above. It has the vectors going across the rows, so there are three vectors, and four tuples. The various parts of the file are labelled, and will be described below:

```
TABLE
0,1
""
VECTORS      Header
0,3          Item
""                      Header
TUPLES
0,4
""
DATA
0,0
""
-1,0
BOT
0,100        1980 Sales
V
0,80         1980 Cost
V
0,20         1980 Profit
V
-1,0
BOT
0,110        Data
V            Value
0,88                    Tuple
V
0,22
V
-1,0                                Data
BOT                                 Part
0,121
V
0,97
V
0,24
V
-1,0
BOT
0,133        1983 Sales
V
0,106        1983 Cost
V
0,27         1983 Profit
V
-1,0
EOD
```

The Header

The header is organized into *header items*. Each header item contains a different piece of information about the data stored in the file. That data is sometimes numeric, and sometimes a string value.

Structure of a header item

Each header item consists of four fields arranged as follows:
Topic
Vector number, Value
"String value"

The Topic

This is a keyword that identifies the header item. It must be a simple token readable as a string in BASIC without quotation marks. A word consisting of just letters with no spaces is best.

The Vector number

Several header items, such as a label, will apply to a specified vector. The Vector number specifies which vector this particular header item refers to. If the header item is not specific to a vector, such as a report title, this value should be 0.

The Value

This appears on the same line as the Vector number. It is used for header items that specify values, such as the number of vectors. It is zero if the value is not used by the header item. The value must be an integer.

The "String value"

This appears on a separate line after the Vector number and Value. It is used for header items that need string values rather than numeric values. The vector labels are an example. The string is always enclosed in quotes.

Thus the header item consists of three lines. The first line is the topic of the header item, the second line consists of two numbers and the third line has a string. The specific header items are described below.

Programs can ignore all header items until one with the topic DATA (described below) is found. The following program segment will skip the header items:

```
1000  INPUT#1,T$           :REM - READ THE TOPIC NAME
1010  INPUT#1,S,N          :REM - READ THE VECTOR #, VALUE
1020  INPUT#1,S$           :REM - READ THE STRING VALUE
1030  IF T$< > "DATA" THEN 1000       :REM - CHECK FOR
                                     :REM - DATA HEADER ITEM
```

The Header Items

The standard header items are shown below with a description. The only required header items are TABLE and DATA, which must be the first and last header items, respectively.

Table **0,version** **"title"**	This is the first entry in the file. While it is not strictly required, it is important to allow for changes in future versions and it allows programs to verify that the file is a TABLE of data. The version number must be 1. Some programs may not accept the file without the TABLE header item.
Vectors **0,count** **""**	This tells how many data vectors are present. Some programs will require this header item to be present. If this header item is absent, the input program can calculate this value by counting the number of Data Values in each tuple (see below). N.B.: This header item must appear before header items that reference vector numbers, such as the LABEL header item.
Tuples **0,count** **""**	Specifies the length of each vector. (All vectors must be the same length.) Some programs will require this header item. If this header item is absent, the input program can calculate this value by counting the number of tuples before an end of data (EOD) Special Data Value (see below).
Label **vector#,line#** **"label"**	Provides a label for the specified vector. This is optional. The line# allows for labels spanning multiple lines, but can be ignored by systems allowing only single line labels. The values 0 and 1 should be equivalent for line#.
Comment **vector#,line#** **"label"**	This is similar to the LABEL header item for systems that allow an expanded description in addition to labels.
Size **vector#,#bytes** **""**	This is used by programs, such as data base systems, that allocate fixed size fields for each value. Such programs, though, should be able to read files that do not contain SIZE information, since other programs may not be able to generate information of this type.
Data **0,0** **""**	This says that data follows. The data is organized by tuples, with one value from each vector in a given tuple.

Subsystems may define their own header items to meet their needs. Header items that will tend to be common should be standardized, such as the LABEL for a vector.

The Data Part

The data part consists of tuples, i.e. one value for each vector, in vector order. The tuples are made up of groups of two numeric values and one string value called *Data Values*. Each Data Value is used to represent the value of one element of data in the file.

In addition to the Data Values used to represent the actual data in the file, there are two types of Special Data Values used to provide information about the organization of the data. One Special Data Value is used to show where each tuple starts, and the other Special Data Value is used to indicate the end of all of the data in the file.

Data Values are all in the following format:

> Type Indicator, Number Value
> String Value

The first two fields are numeric values on a single line, the last is a string on a line by itself. These fields are:

The Type Indicator field

The Type Indicator is an integer that is used to indicate the way in which to interpret the rest of the fields in a Data Value. The currently assigned values for the Type Indicator are:

−1 Indicates that this Data Value is a Special Data Value, either a beginning of tuple indicator or an end of data indicator. See below for a discussion of the Special Data Values.

0 The data is numeric. The value of the Data Value is stored in the Number field possibly modified by the String Value (see the descriptions of the Number Value and String Value fields below).

1 The data is a string. The value of the Data Value is stored in the String Value field.

2 This is an application specific value. The meaning is determined by the cooperating programs that are expected to use the data. For example, it might be an expression in the host language. For simple applications these values can be treated as strings.

The Number Value field

This is used when the Type Indicator is 0 to represent the value. The value must be a decimal (base 10) number. It may optionally be preceded by a sign (+ or −), have a decimal point, and immediately be followed by the letter E and an optionally signed power of ten exponent. The number may be preceded or followed by one or more blanks. Note that this is the only place in DIF where a non-integer value is allowed. Some programs that read data in DIF may only accept integer values (e.g., programs written in some BASICS or some systems programming languages).

The String Value field

The interpretation of this field depends upon the Type Indicator.

For normal Type Indicator 0 (numeric) data, the String Value should be the letter V (for value). If it is not V, then it is a Value Indicator, used to override the value. A subsystem may choose its own Value Indicators for named values, though they should be registered with the DIF Clearinghouse. The following Value Indicators are used by VisiCalc:

> V
> This is the normal case for numbers.
>
> NA
> This is a value marked explicitly as Not Available. The Number Value is set to 0.
>
> ERROR
> This is a value that represents the result of an invalid calculation, such as division by 0. The Number Value is set to 0.

It should always be possible to ignore the String Value for numeric data and just use the Number Value given. Another simple approach is to treat all values with a Value Indicator other than "V" as missing. Note that quotes are not permitted around the Value Indicator (for the sake of some BASICS).

For the Type Indicator of 1 (string data), this field is used for the string value itself. The quotes are optional if the field consists of just letters and does not contain any spaces. However, if a starting quote is given, a terminating quote must also be given.

Each tuple begins with a Special Data Value whose Type Indicator is −1, Number Value is 0, and whose String Value is BOT (for Beginning Of Tuple). This Special Data Value can be used by programs to determine how many vectors are in the file in the absence of a VECTORS header item (by counting the number of Data Values between BOT Special Data Values), or for a program to verify its position in a file.

At the end of the last tuple is a Special Data Value with a Type indicator of −1, a Number Value of 0, and a String Value of EOD (For End Of Data). This will allow programs to determine the number of tuples in the absence of a TUPLES header item (by counting the number of tuples before an EOD Special Data Value), and to gracefully detect the end of the file.

4. Sample Programs

Here are two sample programs. The first program creates a DIF file. The second program can read a DIF file and list its contents. They should be helpful in understanding how to manipulate DIF files. They are written as main programs with subroutines, so you can pick up code from them to be used in other programs.

Creating a DIF File

```
100  REM - This program creates a Data Interchange Format file.
110  REM - It prompts for the file name, number of vectors and
120  REM - tuples, and then for the values themselves.  Data
130  REM - may be either numeric (type 0), or string (type 1).
160  REM
1000 PRINT "FILE NAME";           :REM - Get name of file
1010 INPUT F$
1020 OPEN "O",1,F$                :REM - Open for write
1030 PRINT "NUMBER OF VECTORS";   :REM - Get number of vectors
1040 INPUT NV                     :REM - into variable NV
1050 PRINT "NUMBER OF TUPLES";    :REM - and number of tuples
1060 INPUT NT                     :REM - into variable NT
1070 GOSUB 3000                   :REM - Write out DIF header
1080 FOR I = 1 TO NT              :REM - Get data and output it
1090   T = -1: V = 0: S$ = "BOT"  :REM - Output beginning of tuple
1100   GOSUB 4000
1110   FOR J = 1 TO NV            :REM - Get each data value
1120     PRINT "DATA TYPE FOR VECTOR #";J;", TUPLE #";I;
1130     INPUT T
1140     V = 0: S$ = "V"          :REM - Init values
1150     PRINT "DATA VALUE FOR VECTOR #";J;", TUPLE #";I;
1160     IF T=0 THEN INPUT V
1170     IF T=1 THEN INPUT S$
1180     GOSUB 4000               :REM - Output the Data Value
1190     NEXT J
1200   NEXT I
1210 T = -1: V = 0: S$ = "EOD"    :REM - Output end of data
1220 GOSUB 4000
1230 CLOSE 1
1240 PRINT "FINISHED CREATING DIF FILE ";F$
1250 STOP
3000                              :REM - Routine to write out DIF header
3010 PRINT#1,"TABLE": PRINT#1,"0,1": GOSUB 3500
3020 PRINT#1,"TUPLES": PRINT#1,"0,";NT: GOSUB 3500
3030 PRINT#1,"VECTORS": PRINT#1,"0,";NV: GOSUB 3500
3040 PRINT#1,"DATA": PRINT#1,"0,0": GOSUB 3500
3050 RETURN
3500                              :REM - Routine to write "" (null string)
3510 PRINT#1,CHR$(34);CHR$(34)    :REM - See Appendix on quoted
3520 RETURN                       :REM - strings in Basic, below
4000                              :REM - Routine to write out Data Value
4010 PRINT#1,T;",";V
4020 PRINT#1,S$
4030 RETURN
4040 END
```

Note that if the string values being saved have spaces or special characters, the code at line 4020 should be changed to check for those cases, and add leading and trailing quotes. See the discussion about Quoted Strings in BASIC in the Appendix.

Listing a DIF File

```
100  REM - This program reads a Data Interchange Format file
110  REM - and lists its contents.  The program prompts for
120  REM - the name of the file to be listed.
150  REM
500  DIM T(100)                    :REM - Maximum of 100 vectors
510  DIM V(100)                    :REM - T, V, and V$ hold the
520  DIM V$(100)                   :REM - Type - Indicator, Number
530                                :REM - Value and String Value
540                                :REM - of each element in a tuple
550                                :REM -
1000 GOSUB 5000                    :REM - Call initialization code
1010 GOSUB 6000                    :REM - Read header
1020 FOR I = 1 TO NT               :REM - Read all of the tuples
1030   PRINT "VALUES FOR TUPLE #";I
1040   GOSUB 7000                  :REM - Get a tuple
1050   FOR J = 1 TO NV             :REM - Output each element
1060     IF T(J)=0 THEN PRINT V(J) :REM - Output numeric value
1070     IF T(J)=1 THEN PRINT V$(J):REM - Output string value
1080     NEXT J
1090   NEXT I
1100 CLOSE 1
1110 PRINT "FINISHED LISTING FILE ";F$
1120 STOP
5000                               :REM - Initialization code
5010 PRINT "FILE NAME";            :REM - Get name of file to read
5020 INPUT F$
5030 OPEN "I",1,F$
5040 NV = 0                        :REM - Init counts of vectors
5050 NT = 0                        :REM - and tuples
5060 RETURN
6000                               :REM - Read header, and set NV and NT
6010 INPUT#1,T$                    :REM - Get Topic Name
6020 INPUT#1,S,N                   :REM - Get Vector Number
6030 INPUT#1,S$                    :REM - Get "String Value"
6040 IF T$="VECTORS" THEN 6500     :REM - Check for known header
6050 IF T$="TUPLES" THEN 6600      :REM - items
6060 IF T$="DATA" THEN RETURN      :REM - DATA ends header
6070 GOTO 6010                     :REM - Ignore unknown ones
6500 NV = N                        :REM - Value is number of vectors
6510 PRINT "THE FILE HAS ";NV;" VECTORS."
6520 IF NV<=100 THEN 6010          :REM - If not too many continue
6530 PRINT "TOO MANY VECTORS.  THIS PROGRAM ONLY HANDLES 100."
6540 CLOSE 1
6550 STOP
6600 NT = N                        :REM - Value is number of tuples
6610 PRINT "THE FILE HAS ";NT;" TUPLES."
6620 GOTO 6010                     :REM - Get next header item
```

```
7000                               :REM - Get all vector elements in a tuple
7010 GOSUB 8000                    :REM - Get next Data Value
7020 IF T1<>-1 THEN 9000           :REM - Must be BOT or else error
7030 IF S$<>"BOT" THEN 9000
7040 FOR K = 1 TO NV               :REM - Get each Data Value
7050   GOSUB 8000
7060   IF T1=-1 THEN 9000
7070   V(K) = V1                   :REM - Save Values and Type
7080   V$(K) = S$                  :REM - Indicator
7090   T(K) = T1
7100   NEXT K
7110 RETURN
8000                               :REM - Get next Data Indicator
8010 INPUT#1,T1,V1                 :REM - Get Type Indicator
8020 INPUT#1,S$                    :REM - Numeric Value and String
8030 RETURN                        :REM - Value
8040                               :REM - Error Processing
9000 PRINT "ERROR IN FILE FORMAT."
9010 CLOSE 1
9020 STOP
9030 END
```

Please note that while the above program can read many DIF files correctly, it depends upon the TUPLES and VECTORS header items to determine the organization of the file. A more general program could be written that, in the absence of these header items, deduced their values from the placement of BOT and EOD Special Data Values. While most programs that deal with DIF should be able to produce TUPLES and VECTORS header items (VisiCalc, for example, does), some may not (such as a program that records data incrementally and doesn't know how many data points it will encounter until it is finished).

5. Appendices

Quoted Strings in BASIC

Writing quoted strings is not always convenient in BASIC. For TRS-80 Basic the CHR$ function must be used. For example:

```
PRINT#1,"TABLE"
PRINT#1,"0,1"
PRINT#1,CHR$(34);"STOCK PRICES FOR ABC COMPUTER CO.";CHR$(34)
```

Character Sets

The character set is assumed to be that of the host machine. Thus, if one is transferring a file from a machine using ASCII to one using EBCDIC, the appropriate conversions must be made. In addition, some machines may require that the quote be changed to an apostrophe. These changes should be transparent to most users. In order to assure compatability, strings should not contain nonprinting characters, other than the end of line sequence (RETURN, C/LF, NEWLINE or whatever).

The ASCII character set defines 95 printable characters. The user should be aware that some systems do not make it easy to use the full set. In particular, keywords (including topic names and number types) must be in upper case. Some systems only support a limited set of characters, often 64 printable characters or less. When transporting a file to such a system the upper and lower case characters would be mapped together to one case. Other special characters may be mapped into common characters. If these transformations affect the integrity of the data, it should be specified in the documentation associated with the data.