COMPU-SHEET
operator guide
for financial planning

88A00759A04

ZEBRA
FAMILY

RECORD OF REVISIONS

Title: COMPU-SHEET Operator Guide For Financial Planning

Document No. 88A00759A04

Date	Revision Record
May 83	Original Issue
Sep 83	Revision B
Mar 84	Revision A03
Feb 85	Revision A04

THIS DOCUMENT CONTAINS PROPRIETARY INFORMATION WHICH SHALL NOT BE REPRODUCED OR TRANSFERRED TO OTHER DOCUMENTS OR DISCLOSED TO OTHERS OR USED FOR MANUFACTURING OR ANY OTHER PURPOSE WITHOUT PRIOR WRITTEN PERMISSION OF GENERAL AUTOMATION, INC.

THIS MANUAL AND THE COMPU-SHEET SOFTWARE IS LICENSED TO THE USER FOR USE ON THE LICENSEE'S COMPUTER SYSTEM AND CONSTITUTES PROPRIETARY INFORMATION AND TRADE SECRETS OF INTERACTIVE SYSTEMS, PHOENIX, ARIZONA. NEITHER THIS MANUAL NOR THE COMPU-SHEET SOFTWARE CAN BE COPIED, SOLD, RENTED, LEASED, GIVEN, ASSIGNED, OR OTHERWISE MADE AVAILABLE TO ANY OTHER INDIVIDUAL OR ORGANIZATION WITHOUT THE WRITTEN CONSENT OF INTERACTIVE SYSTEMS.

ALGEBRAIC FORMULA FORMAT
expression consists of:

BASIC ARITHMETIC OPERATIONS: + - * /

LOCATIONS:

col.row - absolute col and row reference
col. - absolute col, on current row
.row - on current col, absolute row
(col or row can be in "Pn" or "Fn" format where "P" = previous, "F" = following, and "n" = number of previous or following columns or rows)

LITERALS: must be enclosed in double quotes and the decimal point must be indicated.

FUNCTIONS: (must be enclosed in single quotes)

'A,loc1,loc2,n' - accumulate (sum)
'AVG,loc1,loc2,S,n' - average
'CNT,loc1,loc2,S,n' - count
'MIN,loc1,loc2,S,n' - minimum value
'MAX,loc1,loc2,S,n' - maximum value
'READ filename id amc' - reads from database
'GET amc' - retrieves another amc
'READW name loc' - reads another spreadsheet
'GETW loc' - retrieves another location
'FIND,floc,sloc,eloc,offset,name' - lookup
'FC,s,n,r,f,p,a,c' - financial calculation

RELATIONAL OPERATORS and CONDITIONAL STATEMENTS:

= equal # not equal < less than > greater than
[less than or equal] greater than or equal
IF condition THEN expression1 ELSE expression2

OTHER OPERATIONS:

: -concatenation D -date T -time
(expression)(conversion) -output conversion
(expression)(I,conversion) -input conversion
(expression)[start,end] -text extract
RD(exp,n) -rounds "exp" to "n" decimal places
SIN(exp) COS(exp) TAN(exp) LN(exp) EXP(exp)
SQRT(exp) PWR(exp1,exp2) REM(exp1,exp2)
ABS(exp) INT(exp) RND(exp)

FORMULA EDITOR:

DF -display R/old/new/n -replace C -copy

18A00289A01 1/85

COMPU-SHEET

QUICK REFERENCE GUIDE

GENERAL AUTOMATION
1045 SOUTH EAST STREET P.O. BOX 4883
ANAHEIM, CALIFORNIA 92803

HELP FEATURE

COMPU-SHEET provides an extensive HELP feature which can be called at ANY prompt. Enter a ? for detailed information about the specific prompt.

ENTERING COMPU-SHEET

Welcome to COMPU-SHEET

File name: - Enter the name of a file designated for spreadsheets.

Spreadsheet name: - Enter the name of the spreadsheet to create/recall. Enter /LIST or /LISTP for a list of spreadsheets in file.

Password: - Enter the password required to access this spreadsheet.

If a new spreadsheet:

Description: - Enter up to 10 lines of desc.

Enter initials: - Enter your initials/name or a /X to go back to File name.

Enter Y or (CR): - Enter a Y to define your own spreadsheet or (CR) to use a standard spreadsheet (14 cols)

(CR) = Carriage Return

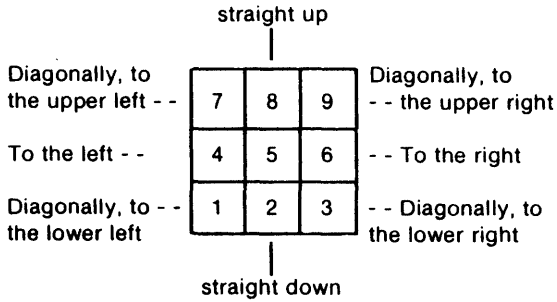
If an existing spreadsheet:

Change description: - Enter a Y to change desc or N or (CR) to continue.

Enter initials: - Enter your initials/name or a /X to go back to File name.

The spreadsheet will display on the screen.

CURSOR CONTROL - with the 10-key keypad



5 key = direct movement to a location/spreadsheet

COMPU-SHEET COMMANDS -

. or , - Data/formula entry into one location (.) or a series of locations (,). A backslash (\ ) at the request for data or formula clears the location.

/AUDIT - Prints a spreadsheet audit report on the system line printer.

/AUDITT - Prints audit report on "slave" printer.

/CAL - Calculates all spreadsheet formulas.

/CALR - Calculates a range of formulas.

/CDIR - Toggles the calculation direction.

/CLR - Clears a range of data/formulas.

/COPY - Copies data/formulas from a col or row to other cols or rows.

/DEL - Deletes columns or rows.

/DIR - Toggles the data entry direction.

/DIS - Redisplays the spreadsheet.

/FI - Files/deletes and exits the spreadsheet.

/FILBLD - Creates data file from the spreadsheet.

/FS - Files the spreadsheet without exit.

/GO or 5 - Moves the cursor to a specific location or "jumps" to another spreadsheet.

/IFORM - Toggles internal/external data format.

/INS - Inserts columns or rows.

/JUST - Defines column justification and mask.

/MACRO - Records a series of commands which can be executed later by name.

/MERGE - Merges all/part of another spreadsheet into the current spreadsheet.

/NEXT - Controls next location for data entry.

/PAGE - Toggles screen display mode.

/PRINT - Prints spreadsheet on line printer.

/PRINTT - Prints spreadsheet on "slave" printer.

/PROT - Protects cells against accidental entry.

/REP - Repeats one location to other locations.

/SEL - Selects the active window.

/SET - Defines/deletes a single window.

/SETH - Locks column headings on screen.

/SIZE - Displays spreadsheet size on screen.

/SORT - Sorts spreadsheet by row.

/SORTB - Sorts any block on spreadsheet.

/SUPP - Suppresses window row number display.

/TERM - Defines terminal and printer settings.

/WIDTH - Redefines column widths.

/WINDOW - Defines a set of vertical windows.

/X - Exits the spreadsheet without filing.

COMPU-SHEET
operator guide
for financial planning

88A00759A04

Copyright © by General Automation, Inc.
1045 South East Street P.O. Box 4883
Anaheim, California 92803
(714)778-4800 (800)854-6234
TWX 910-591-1695 TELEX 685-513

RECORD OF REVISIONS

Title: COMPU-SHEET Operator Guide For Financial Planning

Document No. 88A00759A04

Date	Revision Record
May 83	Original Issue
Sep 83	Revision B
Mar 84	Revision A03
Feb 85	Revision A04

THIS DOCUMENT CONTAINS PROPRIETARY INFORMATION WHICH SHALL NOT BE REPRODUCED OR TRANSFERRED TO OTHER DOCUMENTS OR DISCLOSED TO OTHERS OR USED FOR MANUFACTURING OR ANY OTHER PURPOSE WITHOUT PRIOR WRITTEN PERMISSION OF GENERAL AUTOMATION, INC.

THIS MANUAL AND THE COMPU-SHEET SOFTWARE IS LICENSED TO THE USER FOR USE ON THE LICENSEE'S COMPUTER SYSTEM AND CONSTITUTES PROPRIETARY INFORMATION AND TRADE SECRETS OF INTERACTIVE SYSTEMS, PHOENIX, ARIZONA. NEITHER THIS MANUAL NOR THE COMPU-SHEET SOFTWARE CAN BE COPIED, SOLD, RENTED, LEASED, GIVEN, ASSIGNED, OR OTHERWISE MADE AVAILABLE TO ANY OTHER INDIVIDUAL OR ORGANIZATION WITHOUT THE WRITTEN CONSENT OF INTERACTIVE SYSTEMS.

FOREWORD

The COMPU-SHEET financial planner is a powerful processor for use under the PICK Operating System, a part of the General Automation ZEBRA series.

This document is provided for user guidance in the use of COMPU-SHEET. Other ZEBRA documentation that is available to the user from General Automation:

Document No.	Title
88A00751A	Overview of the PICK Operating System
88A00757A	PICK Operator Guide
88A00758A	ACCU-PLOT Operator Guide
88A00760A	Quick Guide for the PICK Operating System
88A00774A	PICK Utilities Guide
88A00776A	PICK ACCESS Reference Manual
88A00777A	PICK SPOOLER Reference Manual
88A00778A	PICK BASIC Reference Manual
88A00779A	PICK EDITOR Reference Manual
88A00780A	PICK PROC Reference Manual
88A00781A	PICK RUNOFF Reference Manual
88A00782A	Introduction to PICK TCL and FILE STRUCTURE
88A00783A	PICK JET Word Processor Guide

TMACCU-PLOT is a trademark of ACCUSOFT Enterprises

TMPICK is a trademark of PICK Systems, Inc.

TMZEBRA is a trademark of General Automation, Inc.

TMCOMPU-SHEET is a trademark of Raymond-Wayne Corporation

--TABLE OF CONTENTS--

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - INTRODUCTION TO COMPU-SHEET - -

COMPU-SHEET is a powerful financial planning tool which can be used by anyone. It does not require programming experience. COMPU-SHEET gives you the ability to solve, in seconds, involved problems which might take hours to do by hand. With COMPU-SHEET you can create virtually any type of spreadsheet for any application. It can be used by controllers, treasurers, CPA's, managers, or anyone who needs to solve financial problems or work with projections.

COMPU-SHEET is an electronic replacement for such traditional financial modeling tools as the accountant's columnar pad, pencil, and calculator. Spreadsheets make it easy for anyone to create, edit, and use financial models. COMPU-SHEET is similar to an accountant's columnar pad, except that the electronic spreadsheet is much larger. Each column is assigned a number and each row is assigned a number. The intersections of the columns and rows are called LOCATIONS or CELLS. Locations are identified by the numbers of the intersecting columns and rows (for example, location (or cell) "3.17" indicates the point where column 3 intersects with row 17). Any cell can be filled with some type of information. This information can be some form of description (such as: "AMOUNT", "SALES THIS MONTH", "WESTERN REGION", "BOB JONES"...), or numeric information (such as: "12,345.67", ".005", "543"...), or can be a formula which will calculate a result based upon information contained on the spreadsheet or somewhere else in your computer (such as: "3.17+4.25" which adds the cell at 3.17 to the cell at 4.25). The real beauty of COMPU-SHEET is that YOU define the problem, YOU design the spreadsheet (or solution), and YOU calculate and print the results. Did you make a spelling error? Correct it on the spot. Is your formula incorrect? It's easy to fix. Would you like to try out several assumptions? It takes only a few seconds to make the change and calculate a new approach to your problem.

COMPU-SHEET can handle problems that are as simple as a columnar spreadsheet or as complex as an involved investment analysis. COMPU-SHEET offers the ability to access any file in your system for retrieval of data needed for either display or computations. You can consolidate multiple spreadsheets into a single summary report.

Your imagination and a good understanding of the operation of COMPU-SHEET is all that is required to put it to work for you. This manual will explain the features and operation of COMPU-SHEET. It is suggested that you start with a simple spreadsheet which you can use for exploring the commands which control the entry and display. You should experiment with the formulas by entering a number of different calculations. Try to use as many of the formula commands as possible, just for practice. The time and effort you put into understanding all the capabilities of COMPU-SHEET will pay off later when you need to use it for a practical application.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

COMPU-SHEET has an extensive built-in "help" feature. YOU CAN ENTER A QUESTION MARK (?) AT ANY POINT DURING OPERATION AND RECEIVE A HELP "WINDOW" on your screen which displays one or more pages of explanation pertaining specifically to the area where you need assistance. This comprehensive "help" feature is like having a tutor standing beside you, ready to give you assistance whenever you ask. It includes comprehensive explanations of the spreadsheet concept, screen layouts, all commands, how to enter data, and how to build formulas.

You can obtain help at any prompt when creating a new spreadsheet.

You can obtain help at any point when recalling an existing spreadsheet.

You can obtain help at the "Command:" prompt for a comprehensive explanation of the spreadsheet layout, cursor movement, entering data, and entering formulas.

You can obtain help at the prompt requesting data.

You can obtain help at the prompt requesting formulas.

You can obtain help at any point during a command sequence.

YOU CAN OBTAIN HELP ANY TIME YOUR TERMINAL IS WAITING FOR A RESPONSE!

Be sure to take advantage of this feature, it will eliminate most references to this manual.

You will find COMPU-SHEET is one of the most useful management tools you have available. The following sections discuss the operation of COMPU-SHEET in detail.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - TERMINOLOGY - -

Before we begin discussing COMPU-SHEET, there are a few terms which will be used in this manual which should be reviewed. We have tried to stay away from "computer" terminology, however some of it is necessary.

<CR> "Carriage Return" - the "RETURN", "ENTER", or "NEW LINE" keys on your keyboard.

CELL Is the point where a column and a row intersect on your spreadsheet. A cell can contain any type of information or a formula. Also called a "location".

COMMAND One or more keystrokes which COMPU-SHEET interprets as an instruction to perform some operation.

CONCATENATE Means "to link together" or "connect". If we concatenate "ABC" and "XYZ" we would have "ABCXYZ".

CURSOR A cursor is an indicator, on the surface of your screen, marking the point where information may be entered onto the screen. Sometimes it is an underline, and sometimes it is a small block which may or may not blink on and off. In COMPU-SHEET, a "window cursor" is either a block or a series of periods (.....) which mark the cell or location which is currently being "pointed to".

FILE, A Means a named area on the disk of the computer which can hold or "store" your spreadsheets.

FILE, TO Is the process of placing one of your spreadsheets into a file. This is also called "storing", "filing" , or "saving" a spreadsheet.

FORMULA Instructions you enter into COMPU-SHEET to compute a value.

JUSTIFICATION Indicates the margin alignment of the information within a column. The information may be aligned with the left margin (L) or with the right margin (R). Usually alphabetic information is left justified and numeric information is right justified. In COMPU-SHEET, the term usually refers to both the justification and the "mask" (defined below).

LOCATION Is the point where a column and a row intersect on your spreadsheet. A location can contain any type of information or a formula. Also called a "cell".

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

MACRO	A series of commands which are given a name. The entire series of commands may be executed by the assigned name.
MASK	The instructions you give specifying the format of the display of the data in a column. The "MASK" indicates the number of decimal places to display, if commas are to be placed between thousands positions, the way credit values will be indicated, etc.
NULL	Means "nothing" or "no value". A "null" response means you entered nothing but a <CR> to a request for input.
NUMERIC KEY PAD	Is the 10-key keyboard on your terminal. It is used to control the movement of the "window cursor" in COMPU-SHEET.
PROMPT	The process of requesting a response from the terminal operator.
REVERSE VIDEO	Refers to a terminal feature which allows a line or part of a line to be displayed in a highlighted format.
STEP VALUE	Indicates the frequency of occurance. A step value of "2" means "every other time", a step value of "3" means "every third time", etc.
TCL	Is an abbreviation for "Terminal Control Language" and refers to the operating system prompt requesting a command. This request is indicated by the character ">" on some systems and ":" on others.
WINDOW	A section of your screen that displays part of a spreadsheet. If your screen is displaying one window, the entire screen is displaying one portion of your spreadsheet. If your screen is split into two windows, part of your screen is displaying one section of your spreadsheet and the rest of the screen is displaying another section, etc.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - ENTERING COMPU-SHEET - -

To activate COMPU-SHEET you must first make sure you are in an account which has been set-up to run COMPU-SHEET, then from TCL enter: COMPU-SHEET

The screen will erase and the following will display:

Welcome to COMPU-SHEET

File name:

All COMPU-SHEET spreadsheets are stored in files. They should be organized so that related spreadsheets are stored together in the same file. Here, you must enter the name of the file where the spreadsheet is (or will be) stored. If the file name entered is not a valid file, the program will display: -INVALID- and will repeat the request for the file name. If the file name is a valid file, the program will display:

Spreadsheet name:

Each spreadsheet you create must be named. This name is assigned at the time the spreadsheet is initially created and is used to "recall" it for all future reference and use. Here you should enter the spreadsheet name. This name must not contain any spaces and cannot be longer than 15 characters. If you would like the screen to display a list of the spreadsheets in this file, enter "/LIST" at this point, or if you would prefer a printed listing, enter "/LISTP". The list will contain all of the spreadsheet names and their descriptions. After the spreadsheet name is entered, the program will display:

Password:

Each spreadsheet may be "password encoded". This means that a special "password" can be assigned to any spreadsheet, making unauthorized access more difficult. This password will not display as it is entered. If no password is desired on a new spreadsheet, you can bypass it by entering <CR> (carriage return) or the RETURN, ENTER, or NEW LINE key. When calling up a previously entered spreadsheet that was password encoded, the correct password must be entered in order to proceed.

PLEASE NOTE THAT YOU CAN OBTAIN HELP AT ANY POINT BY ENTERING A "?". Once "help" has been requested, you will continue to receive help until the command or process has been completed. This "help" feature will save you the effort of constantly referring to this manual while you are becoming familiar with COMPU-SHEET.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

At this point, one of two things will happen, depending on whether this is a new COMPU-SHEET spreadsheet or not.

If it is a previously entered spreadsheet, the screen will display the description and will ask if you want to change it. If you do want to make a change, answer "Y" and you can reenter the description. If not, enter <CR> and COMPU-SHEET will display the initials of the last person to update this spreadsheet, and the date and time of the last update. It will then ask you to enter your initials. Enter them followed by a <CR> and the spreadsheet will be displayed exactly as it was the last time it was filed. In this case, you are ready to enter any of the commands which control COMPU-SHEET.

If it is a new spreadsheet, it is necessary to enter a description, your initials, and to define the initial size and format of your spreadsheet. You may choose to utilize a predefined format of 14 columns or you may define your own format.

The following discussion will take you thru the set-up of a new spreadsheet. You will be prompted:

*** New spreadsheet ***

Description:

Each new spreadsheet must have a "description" associated with it. This description can consist of up to 10 lines of 60 characters each. The purpose of the description is to provide a record of the reason a spreadsheet was created and information concerning usage and retention. It may contain any descriptive information you desire. The program will then ask:

Please enter your initials/name or "/X":

Here, you must enter either your initials or name (up to 10 characters) or a "/X". If you enter "/X", the program will return to the spreadsheet name prompt. If you enter your initials or name, the spreadsheet will clear the screen and display:

New spreadsheet - file name: (name) spreadsheet name: (name)

Enter "Y" if you wish to define your own spreadsheet format or <CR> to use standard spreadsheet format:

COMPU-SHEET provides a standard spreadsheet format consisting of 14 columns defined as follows:

Column 1 is 15 characters wide and data displays left justified (L). Columns 2 thru 14 are 10 characters wide, right justified and will display numbers with 2 decimal places and commas between the thousands positions (R2,).

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

If you prefer to define your own format, enter "Y". If you prefer to use the standard format, enter a <CR>. To cancel the entire process and start over, enter an "X".

In either case, once you are in the spreadsheet you have the option to insert, add or delete columns or rows.

If you indicate you wish to use the standard format, the screen will erase and a blank spreadsheet will display. At this point, you will be sitting at the "Command:" prompt and can begin to enter data and formulas into the spreadsheet. You can skip to the next section of this manual.

If you indicate you want to define your own format, the following will display:

```
.........Define spreadsheet size and format.........
From COL      Thru COL        WIDTH        JUSTIFICATION
      1
```

This display begins the process that will allow you to define the number, width and format of the columns to be created for your spreadsheet. The first prompt is the "From COL" and is always filled in by COMPU-SHEET.

The second prompt is "Thru COL" and represents the ending column in the range being defined. This range begins with the column number listed under the "From COL" and ends with the number entered here. The entry of a <CR> here indicates the end of the definition process.

The third prompt is the "WIDTH" and represents the width (in characters) to be assigned to each column in the range. The width of a column may range from 1 to 75.

The last prompt is the "JUSTIFICATION" and defines the format of the display. A discussion of justification can be found under the definition of the command "/JUST" in this manual. In short, the justification indicates whether the information in the column is to be left (L) or right (Rxxxx) justified, and if right justified, how many decimal places should appear - if commas are to appear between thousands positions - the type of credit indication (such as: <1,234.56> or 1,234.56CR or 1,234.56- etc...). If you enter a "?", several examples of justification will be displayed as part of the help process.

Once all of the columns have been defined, enter <CR> at the "End COL" prompt and the blank spreadsheet will be displayed. The next section will discuss the spreadsheet format.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - THE SPREADSHEET FORMAT - -

The layout of the spreadsheet should be clearly understood in order to effectively utilize COMPU-SHEET. The following is a typical new spreadsheet display:

```
**********************************************************************************
*    .....1].....2].....3].....4].....5].....6].....7].....8]    *
*  1 .....                                                       *
*  2                                                             *
*  3                                                             *
*  4                                                             *
*  5                                                             *
*  6                                                             *
*  7                                                             *
*  8                                                             *
*  9                                                             *
* 10                                                             *
* 11                                                             *
* 12                                                             *
* 13                                                             *
* 14                                                             *
* 15                                                             *
* 16                                                             *
* 17                                                             *
* 18                                                             *
* 19                                                             *
* 20                                                             *
* Loc: 1.1      Width: 12      Just: L      Dir: R;R;P     Window: 1       *
* Data:      Formula:                                            *
* Command:                                                       *
**********************************************************************************
```

Let's examine this layout.

The first row, consisting of a series of periods followed by a number and a right bracket "]", indicates the columns on your spreadsheet.

The column of numbers ranging from one to twenty on the left side of the screen are the row indicators. Any group of 20 contiguous rows can be displayed at one time.

The point where a column intersects with a row is referred to as a "LOCATION". A location may contain heading information, data of any type, or the results of any computation.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

The two lines ("Loc:" and "Data:") are the status lines. These lines will keep you informed of a number of things which you may need to know while you are using COMPU-SHEET.

Loc: - indicates the current location of the cursor. In the example, the cursor is located at 1.1. That is, the cursor is set at the location where column 1 intersects with row 1. Throughout COMPU-SHEET, all locations are referred to by column and row. Locations are entered as: "col.row" where "col" is the column and "row" is the row. Therefore, column 7, row 5 would be: "7.5"; column 17, row 12 would be: "17.12"; etc... If the location is followed by an asterisk "*", it is an indication that the location is "protected" against data or formula entry. See the discussion on the "/PROT" command for more information.

Width: - indicates the defined width of the locations within the current column. In the example, the location at the intersection of column 1, row 1 has a width of "12". (As every location in Column 1 has a width of "12".)

Just: - indicates the justification and mask of the current column. In the example, column 1 has a justification of "L". Other columns may have a justification such as "R2,", which means the data is to be right justified (aligned with the right margin) and any numeric data should be displayed with 2 decimal places and commas between the thousands positions. See the discussion of the "/JUST" command for more information.

Dir: - displays the status of 3 indicators separated by semicolons ";". They represent the following:

The first indicator represents the direction the cursor will move after entering data or formulas. "R" indicates the cursor will move from left to right across a row. "C" indicates the cursor will move from top to bottom down a column. See the "/DIR" command for further explanation.

The second indicator represents the direction of the calculation process. "R" indicates the calculation proceeds across each row, starting with row 1. "C" indicates the calculation proceeds down each column, starting with column 1. See the "/CDIR" command for further explanation.

The third indicator displays the setting of the "page" indicator. As the cursor reaches a screen margin, COMPU-SHEET will either display the next screen "page" (indicator set to "P") or will display the next column or row (indicator set to "N"). See the "/PAGE" command for further explanation.

There may be a fourth display in this section. The command "/NEXT" provides for indicating the "next" location to receive data or formula entry. See the discussion of the "/NEXT" command for more information.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

Window: - indicates the current window selection. Any spreadsheet can have the screen split into two or more windows. The cursor can move only within the current window selection. If you would like to move the cursor into another window, it can be selected with the "/SEL" command. See the discussion on windows for more information.

Data: - displays the data as it exists in the location of the cursor.

Formula: - displays any formula associated with this location. Formulas are discussed in detail later in this manual.

The last line on the screen is the Command: line. This is the line where the user is prompted for all COMPU-SHEET commands. The next section explains all of the commands available for controlling the operation of COMPU-SHEET. REMEMBER, if you enter a "?" at the Command: line, all valid commands will be displayed.

Notice location 1.1 of our example contains ".....". This is COMPU-SHEET's "window cursor". If you are operating on a terminal which has no reverse video feature, this cursor will consist of a series of periods the width of the column. If you are operating on a terminal with a reverse video feature, this cursor will display in the reverse video format. When you enter a command that causes cursor movement, the cursor will move from the current location to the location selected.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - COMPU-SHEET COMMANDS - -

COMPU-SHEET is controlled by a series of commands which govern such things as cursor movement, changing column widths and justification, copying, window manipulation, storing, printing, and so on. The following sections discuss in detail these commands and their effect. The user should carefully review each of these commands to obtain an understanding of the power of COMPU-SHEET.

When either a new spreadsheet or a spreadsheet containing previously stored data is displayed on the screen, COMPU-SHEET is ready to accept a command at the Command: prompt. After a command is entered, it will be executed and COMPU-SHEET will return to the "Command:" prompt, waiting for another command.

All commands (except cursor movement, "=macro", and the "." & "," entry commands) require a "/" as the first character of the command. COMPU-SHEET will examine the first command character entered for the "/" and if it is found, the rest of the command can be input. If the "/" is not found, a check will be made for the commands 1-9, the "=macro" command, and the "." or "," entry commands. If found, these will be executed. Otherwise, COMPU-SHEET will "beep" and will return to the Command: prompt.

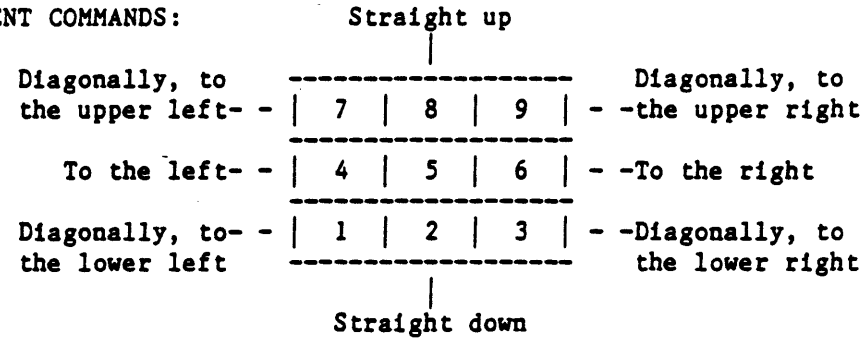
DON'T FORGET, at any point you can enter a "?" for help. COMPU-SHEET will display an explanation of what it expects you to enter.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - COMMAND SUMMARY - -

CURSOR MOVEMENT COMMANDS:



COMPU-SHEET COMMANDS:

.	-	activates the data/formula entry process for one location
,	-	activates the data/formula entry process for a range of locations
/AUDIT	-	prints a spreadsheet audit report on the line printer
/AUDITT	-	prints a spreadsheet audit report on a terminal "slave" printer
/CAL	-	calculates all formulas on the spreadsheet
/CALR	-	calculates the formulas in a specified range of locations
/CDIR	-	changes the direction of execution of formulas
/CLR	-	clears data and/or formulas in a range of locations
/COPY	-	copies information in a column or row to other columns or rows
/DEL	-	deletes columns or rows from the spreadsheet
/DIR	-	changes the direction of the prompting for data and formulas
/DIS	-	redisplays the spreadsheet on the screen
/FI	-	stores the spreadsheet into the file and exits the spreadsheet
/FILBLD	-	creates a file from a spreadsheet
/FS	-	stores the spreadsheet into the file without exiting the spreadsheet
/GO or 5	-	moves the cursor to a specified location in the spreadsheet
/IFORM	-	displays the internal format of the data
/INS	-	inserts columns or rows into a spreadsheet
/JUST	-	redefines the column justification and masking
/MACRO	-	records a series of commands which can be executed later by name
/MERGE	-	merges all or part of another spreadsheet into the current one
/NEXT	-	controls the selection of the next location for data entry
/PAGE	-	controls the display of the next "page" or next column/row
/PRINT	-	prints all or part of the spreadsheet on the line printer
/PRINTT	-	prints all or part of the spreadsheet on a terminal "slave" printer
/PROT	-	protects cells against accidental data or formula entry
/REP	-	repeats the contents of one location to other locations
/SEL	-	selects the active window
/SET	-	defines a single window
/SETH	-	locks column headings on the screen
/SIZE	-	displays the current size of the spreadsheet
/SORT	-	sorts rows of data in the spreadsheet
/SORTB	-	sorts a block of spreadsheet locations
/SUPP	-	suppresses the display of window row numbers
/TERM	-	defines printer and terminal settings
/WIDTH	-	redefines column width
/WINDOW	-	defines a set of vertical windows
/X	-	exits the spreadsheet without writing it into the file

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - CONTROLLING THE CURSOR WITH THE 10-KEY PAD - -

Cursor movement is controlled by the numeric keys (except 0). If you will look at your numeric keypad, notice the relative positions of the numbers 1 to 3, 4 and 6, and 7 to 9. These keys will cause the cursor to move in the same relative direction. The 5 key is equivalent to the "/GO" command explained in the next section.

KEY	DIRECTION OF CURSOR
1	Diagonally, to the lower left
2	Straight down
3	Diagonally, to the lower right
4	To the left
5	To a specified location or another spreadsheet
6	To the right
7	Diagonally, to the upper left
8	Straight up
9	Diagonally, to the upper right
0	No cursor movement

The "carriage return" is automatic upon entry of these commands.

Upon execution of the cursor movement command, the status lines will display the location of the cursor, the width, justification, data, and formula of the new location.

COMPU-SHEET will allow cursor movement within the following constraints:

TO THE LEFT - to column one.

TO THE RIGHT - to the last defined column.

UPWARD - to row one.

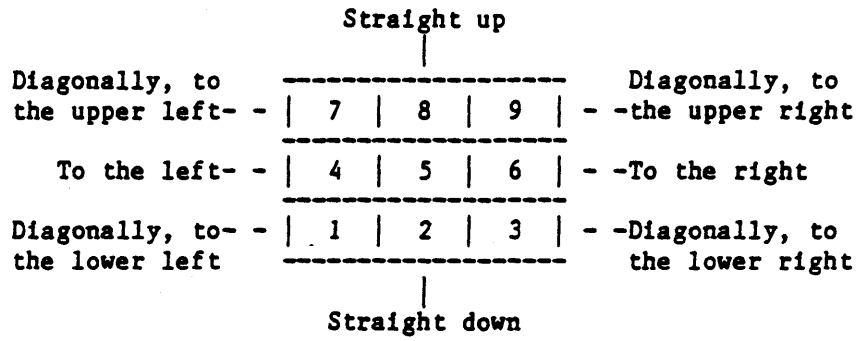
DOWNWARD - there is no downward limit.

If the entered command instructs the cursor to move to a column or row not currently displayed, the entire window will be shifted to "uncover" either the next "page", or the new column or row (the column or row at the opposite side will "roll" off the window). See the "/PAGE" command for more information.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

The following is a diagram of the numeric key pad and the effect of the keys on cursor movement:

THE NUMERIC KEY PAD AND CURSOR MOVEMENT



The "5" key allows you to "go directly to" a specified location or to another spreadsheet.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - MOVING THE CURSOR TO A SPECIFIC LOCATION OR SPREADSHEET = /GO and 5 - -

The "/GO" and "5" commands will direct the cursor to go directly to a selected location or will cause the current spreadsheet to be exited and another called. You can "GO" to any location on the spreadsheet or to any other spreadsheet in the file. Either "/GO" or "5" is acceptable and function alike, although the "5" command does not require a "carriage return".

Upon entry of "/GO" or "5" the command line will request:

Location (col.row) or spreadsheet name:

Here, enter the column, row, or location you want to "go" to or enter the name of the spreadsheet you want to call. The following formats are valid:

col	indicates cursor is to move to the column "col" and remain in the current row.
.row	indicates cursor is to move to the row "row" and remain in the current column.
col.row	indicates cursor is to move to location "col.row".
name{,p}	where "name" is any spreadsheet name in the file and where "p" is the valid password for "name".

In all cases, if the specified column, row, or location is outside the current window, the window will be shifted accordingly.

If an invalid location is entered (outside of the last column), the terminal will "beep" and the command request will be made again.

If a spreadsheet name is entered, the current spreadsheet will be exited and the specified spreadsheet will be displayed. It is important to note that ANY CHANGES TO THE CURRENT SPREADSHEET WILL NOT BE SAVED. Be sure to file the spreadsheet (with the /FS command) first if any changes are to be saved. If the named spreadsheet has a password which is different from the password of the current spreadsheet, the valid password must be entered. If there is no password associated with the named spreadsheet or if the password is identical to the current spreadsheet, the password is not required.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

EXAMPLES:

Command: /GO Location: 8.6
Action: The window cursor will move directly to location 8.6

Command: 5 Location: 1.12
Action: The window cursor will move directly to location 1.12

Command: 5 Location: 24
Action: The window cursor will move to column 24 and remain on the current row. If column 24 was outside the window, the window will be adjusted accordingly.

Command: 5 Location: .37
Action: The window cursor will move to row 37 and remain in the the current column. If row 37 was outside the window, the window will be adjusted accordingly.

Command: 5 Location: BUDGETS
Action: The current spreadsheet will be exited (without saving changes) and the spreadsheet "BUDGETS" will be displayed. The spreadsheet "BUDGETS" either has no password or the password is the same as the current spreadsheet.

Command: 5 Location: PROJECTION,XTX
Action: The current spreadsheet will be exited (without saving changes) and the spreadsheet "PROJECTION" will be displayed. The password for the new spreadsheet is "XTX".

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - ENTERING DATA AND FORMULAS = . (period) and , (comma) - -

There are 2 commands which initiate the input of either data or formulas.

The command "." (period) can be used to instruct COMPU-SHEET to accept the input of data or formulas for a single location. (The "carriage return" will automatically occur when this command is entered).

The "," (comma) command, like the "." command, initiates the input of data and formulas. However, the "," command will prompt for input of data or formulas over a series of locations rather than a single location.

Upon entry of either of the commands the window cursor will display a row of periods and is ready to accept data. You can enter alphabetic characters (in which case the column mask has no effect) or you can enter numeric characters (in which case the column mask will be applied). Data can be entered up to the defined width of the location (in which case a "carriage return" is automatic) or you can enter data less than the defined location width and enter a carriage return. At this point, the data entered will be justified and redisplayed. The window cursor will move to the next location according to the "direction" currently set and you will either be prompted for the next command (if you had entered a ".") or you will be prompted for input of data or formulas for the next location (if you had entered a ",").

A null response (a <CR>) at the request for data, or the existence of a previously entered formula instructs COMPU-SHEET to prompt for a formula. At this point, the command line will display the FORMULA: prompt. You can enter any valid formula such as "3.2+3.4-3.7" (formulas are discussed in detail later). After the formula is entered, the results of the formula will be calculated and will be displayed. The window cursor will then move to the next location according to the "direction" currently set and you will be prompted for the next command. A null response (a <CR>) at the request for a formula will place you back at the Command: prompt.

It is possible to "clear" the contents of any location with the entry commands. The entry of a "\" (backslash) character at the request for data or formula (after entering a "." or ",") will instruct the program to clear the data and formula from the current location.

Note: Numeric entries for columns that have a mask are stored internally with 4 implied decimal places. All numeric entries should be entered with a decimal point indicating the decimal location. The display and calculation processes will maintain accurate decimal locations for you.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

EXAMPLES:

Command: .
Action: The window cursor is ready to accept any data. A null response at this point indicates a formula is to be entered. Upon entry, COMPU-SHEET will return to the Command: prompt.

Command: . FORMULA:
Action: Enter any formula, in the algebraic or function format. Such as: ((3.2/"1.185")*3.5)+3.7 or 'A,.4,.7'
(Refer to "FORMULA FORMATS" for more information.)

Command: ,
Action: The window cursor is ready to accept any data or a null response which indicates a formula is to be entered. Upon entry, the next location will be ready to accept data or formulas.
NOTE: when you have finished entering data and formulas using this command, enter a <CR> twice to return to the Command: prompt.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - CONTROLLING PROMPT AND CALCULATION DIRECTIONS = /DIR, /CDIR, and /PAGE - -

The "/DIR" command will "toggle" (switch back and forth) the prompt direction indicator. The prompt direction indicator is the first value (R or C) separated by a ";" following the "Dir:" display on the status lines. Each entry of this command will change the prompt direction indicator from "R" (row) to "C" (column) or from "C" to "R".

When data or a formula is entered, the window cursor will move to the next location according to the prompt direction indicator. If the indicator is set to "R", the next location selected will be on the same row to the right of the current location (up to the right margin of the spreadsheet). If the indicator is set to "C", the next location selected will be in the same column directly below the current location. If entering a series of data down a column, the indicator should be set to "C"; if entering a series of data across a row, the indicator should be set to "R".

The "/CDIR" commmand will "toggle" the calculation direction indicator. The calculation direction indicator is the second value separated by a ";" following the "Dir:" display on the status lines. If it is set to "R", the calculations will execute across a row, from left to right, starting with row 1 and finishing with the last row. If it is set to "C", the calculations will execute down a column, from top to bottom, starting with column 1 and finishing with the last column.

The direction of calculation can have an important effect upon the results of the calculations. For example, suppose a formula in column 2, row 10 (location 2.10) uses the results of another calculation in column 7, row 3 (location 7.3). If the direction indicator is set to an "R", the results of location 7.3 will be computed before location 2.10 is calculated. The results of this second calculation will be correct. However, if the direction indicator is set to a "C", location 2.10 will be calculated BEFORE location 7.3 is calculated. The results of the calculation at 2.10 would be unpredictable. Therefore, THE CALCULATION DIRECTION INDICATOR CAN HAVE A SIGNIFICANT IMPACT UPON YOUR SPREADSHEET.

The "/PAGE" command will "toggle" the "page" indicator. The "page" indicator is the third value separated by a ";" following the "Dir:" display on the status lines. If the indicator is set to "P", when the cursor moves to a screen margin COMPU-SHEET will display the next "page" of columns or rows. For example, if you have a spreadsheet consisting of 30 columns, and columns 1 thru 7 are currently displayed, when the cursor moves to the right past column 7, columns 8 thru 14 will be displayed and the cursor will be in column 8.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

If the indicator is set to "N", when the cursor moves to a screen margin, COMPU-SHEET will "uncover" the next column or row and the opposite column or row will scroll off the screen. Using the example above, column 8 would display on the right side, column 1 would drop off the left side. Columns 2 thru 8 would be displayed and the cursor will be positioned in column 8.

Note: When entering a new spreadsheet, these direction indicators will be set to "R;R;P". If they are reset, COMPU-SHEET will use the new settings each time the spreadsheet is recalled. If your spreadsheet has been divided into two or more windows, the "PAGE" will automatically be set to "N" to maintain window synchronization.

EXAMPLES:

Command: /DIR
Action: If the prompt direction indicator had been set to a "C", it will be changed to an "R". If the direction indicator had been set to an "R", it will be changed to a "C".

Command: /CDIR
Action: If the calculation direction indicator had been set to a "C", it will be changed to an "R". If the calculation direction indicator had been set to an "R", it will be changed to a "C".

Command: /PAGE
Action: If the page indicator had been set to a "P", it will be changed to an "N". If the page indicator had been set to an "N", it will be changed to a "P".

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - REDISPLAY THE SPREADSHEET = /DIS - -

The "/DIS" command will instruct COMPU-SHEET to redisplay the entire screen according to the current window settings.

Occasionally, if a long formula is entered or if a special terminal control character (such as "clear screen") is accidentally entered, the spreadsheet may "roll" up on the screen one or more lines. This causes the display to align incorrectly with COMPU-SHEET. The use of the /DIS command will redisplay the entire screen and bring it back into alignment.

Upon completion of the display, you will be prompted for the next command.

EXAMPLES:

Command: /DIS
Action: The entry of this command causes COMPU-SHEET to redisplay the screen according to the current window settings.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - CALCULATING THE SPREADSHEET = /CAL and /CALR - -

The "/CAL" command instructs COMPU-SHEET to recalculate and redisplay all locations containing formulas.

The "/CALR" command instructs COMPU-SHEET to recalculate and redisplay a specified range of locations containing formulas.

IT IS IMPORTANT TO NOTE THAT THE /CALR COMMAND WILL RECALCULATE ONLY THE LOCATIONS INDICATED. ANY OTHER SPREADSHEET LOCATIONS DEPENDING UPON THE RESULTS OF ANY LOCATIONS CALCULATED WITH THE /CALR WILL BE IN ERROR IF THE CALCULATED LOCATION VALUES HAVE CHANGED.

Upon entry of the "/CALR" command, the second status line will display:

**** WARNING: Range calculation may yield inaccurate results ****

Then you will be prompted:

Range:

Enter the range of locations to calculate in the format:

col.row-col.row,d{/col.row-col.row,d...}

Where the first location is the upper left location of the "box" of cells to calculate and the second location is the lower right location of the "box" of cells to calculate. The "d" indicates the direction to calculate (and will override the calculation direction indicator setting), and can be either "C" or "R" to indicate "by column" or "by row". For example, a response of:

2.5-2.10,C

would calculate column 2 from row 5 thru row 10. Another example:

2.5-3.10,C/5.7-15.13,R

would calculate columns 2 and 3 from row 5 thru row 10 by column, and then would calculate columns 5 thru 15 from row 7 thru row 13 by row.

This procedure provides the means to calculate a partial spreadsheet and/or the means to calculate a spreadsheet in which part of the spreadsheet must be calculated in the column direction and part must be calculated in the row direction.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

As the spreadsheet is being calculated, you will notice a display to the right of the command. This display is the current location being calculated. As each location is calculated, it will be displayed on the screen.

If the formula processor encounters an invalid formula or an illegal type of operation (such as divide by zero), the screen location will display the message "ERRn", indicating an error situation. The "n" is an error number which will help in identifying the cause of the error. A list of error numbers and descriptions follow:

ERR #	DESCRIPTION
1	Invalid formula format
2	Invalid location format
3	Arithmetic operation attempted on non-numeric data
4	Attempted operation where zero is an illegal operand
5	Invalid "JUMP" command format
6	Attempt to raise a negative number to a non-integer power
7	Multi-valued attribute "SUM" encountered non-numeric data
8	Invalid text extract parameters
9	Can't locate file named in formula
10	Read or write not authorized
11	Read not authorized
12	Write not authorized
13	Can't locate item named in formula in file specified
14	Invalid GOTO function format
15	The GOTO function will be executed by the /CAL command
16	Invalid FIND function format
17	Value to use in FIND function is less than first table entry

EXAMPLES:

Command: /CAL
Action: The entry of this command causes each spreadsheet location containing a formula to be recalculated and redisplayed.

Command: /CALR Range: 5.10-5.20,C
Action: This command will calculate column 5 starting with row 10 and ending with row 20.

Command: /CALR Range: 3.4-8.24,C/10.4-12.24,R
Action: This command will calculate the range of locations from column 3 thru column 8 on row 4 thru row 24 by column, and then will calculate column 10 thru 12, rows 4 thru 24 in the row direction.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - CLEARING DATA AND FORMULAS = /CLR - -

The "/CLR" command provides for clearing the data, formulas, or both from a range of locations.

There are times when you will want to clear some data out of a spreadsheet so that a new set of data can be entered and calculated. The "/CLR" command can be used for this purpose. After entry of the command, the command line will prompt for:

Range:

Enter the range of locations in the format:

col.row-col.row

Where the first location represents the upper left location of the range and the second location represents the lower right location of the range.

Then you will be prompted for:

D,F,B:

The entry of a "D" will clear only the data in the range specified. Entry of an "F" will clear only the formulas in the range. Entry of a "B" will clear both the data and the formulas in the range specified.

For example: If you want to clear the data from columns 2 thru 5 in rows 4 thru 9, the RANGE would be 2.4-5.9 (where 2.4 represents the upper left location and 5.9 represents the lower right location). The entry of a "D" to the "D,F,B" prompt will cause the clearing of only the data within the range of the locations. If you were to enter an "F" to the "D,F,B" prompt, only the formulas would be cleared. A "B" would have caused both the data and formulas to be cleared within the range of the locations.

Note: To clear the contents of just one location, enter a "." as if you are going to enter data, and then enter a "\" (backslash) character. The entire contents of that location (both data and formula) will be cleared.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

EXAMPLES:

Command: /CLR Range: 3.4 D,F,B: D
Action: This command will clear only the data from location 3.4.

Command: /CLR Range: 3.4-8.4 D,F,B: F
Action: This command will clear the formulas from row 4, locations 3.4, 4.4, 5.4, 6.4, 7.4, and 8.4.

Command: /CLR Range: 3.4-3.10 D,F,B: B
Action: This command will clear both the formulas and the data from column 3, locations 3.4, 3.5, 3.6, 3.7, 3.8, 3.9, and 3.10.

Command: /CLR Range: 3.4-5.8 D,F,B: D
Action: This command will clear the data from columns 3, 4, and 5 on rows 4, 5, 6, 7, and 8.

Command: .
You will now be prompted for data or a formula.
The entry of a "\" backslash will clear both the data and formula from the location where your cursor is currently located.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - PROTECTING AGAINST ACCIDENTAL DATA/FORMULA ENTRY = /PROT - -

Once a spreadsheet is created and put into production, there will be times when you will want to protect certain cells against accidental overlay during data entry. The "/PROT" command can be used to "lock" a cell or block of cells so that no one can accidentally overwrite the contents of the cells.

Upon entry of the command, you will be prompted:

Range:

Here, enter the range of cells to protect in the format:

col.row-col.row

Where the first location is the upper left location of the block of cells to protect and the second location is the lower right location of the block.

You will then be prompted:

Set (S) or Clear (C):

Enter an "S" to set the protect in this block or a "C" to clear the protect within the block.

Any protected cells will have an asterisk ("*") display beside the cell number in the "Loc:" on the first status line.

EXAMPLES:

Command: /PROT Range: 2.5 Set (S) or Clear (C): S
Action: This command has set the protect on cell 2.5. No entry of data or formulas may occur until the protect is cleared.

Command: /PROT Range: 2.5-10.25 Set (S) or Clear (C): C
Action: This command has cleared the protect in the block of cells from 2.5 as the upper left location and 10.25 as the lower right location.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - CONTROLLING THE NEXT CELL FOR DATA ENTRY = /NEXT - -

It is common to have a spreadsheet that is updated daily (or weekly, etc.) such as daily cash receipts, sales reports, etc.

When entering data into this type of spreadsheet, there are times it would be convenient to have the cursor automatically move from the bottom of one column to the top of the next without the need to exit the data entry process, move the cursor, and then reenter the data entry mode. The same situation exists when entering data across rows - it would be convenient for the cursor to automatically drop down one row and move back to column one or two. The "/NEXT" command provides this ability.

The "/NEXT" command records in one cell the next location which is to receive data entry. It is important to remember that the "/NEXT" function operates ONLY when in the comma (",") mode of data entry. In order to use this command, move your cursor to the cell where the next location is to be recorded (usually the last cell in a column or row) and enter the "/NEXT" command. You will be prompted:

Enter next location for data or formula entry:

> Here, enter the location where the cursor is to move after data or formula entry in the current location. If only a <CR> is entered, the "next" location will be cleared.

When a cell has a "next" cell number recorded, the location of the "next" is displayed on the first status line with the direction indicators.

EXAMPLES:

In these examples, we will assume the current location is "2.25".

Command: /NEXT Enter next location for data or formula entry: 3.5
Action: Cell 3.5 will be automatically selected for data or formula entry upon completion of the entry of data or formulas in cell 2.25. This command will be effective only when in the "," entry mode.

Command: /NEXT Enter next location for data or formula entry:<CR>
Action: The "next" location for cell 2.25 will be cleared by this command.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - RECORDING AND EXECUTING COMMAND SEQUENCES = /MACRO - -

The "/MACRO" is one of the most powerful commands available on COMPU-SHEET. It enables you to record a series of commands and store them under a name. This recorded sequence of commands can then be executed automatically at a later time by recalling them by name.

This series of recorded commands is called a "macro". The following scenario may help you understand the macro concept:

> A spreadsheet is called up and the command "=DAILY" is entered at the command line. At this point, the cursor moves directly to the location on the spreadsheet where daily sales information is to be entered, pauses for the entry of the data by the operator, then automatically calculates the spreadsheet, prints 3 copies for management, and files the spreadsheet.

You can do this with macro instructions. The following sequence will explain in detail exactly how the "/MACRO" instruction operates:

> The macro begins with the entry of the command "/MACRO". You will then be prompted:
>
> Macro name:
>
> Enter the name you want to call the macro. It should be 10 characters or less. For this discussion, let's pick the name "DAILY". Upon entry of the name "DAILY", status line 2 will display:
>
> Rec macro: DAILY
>
> At this point, EVERY KEYSTROKE IS BEING RECORDED. Even your mistakes. Now you can move the cursor to any location and enter any command (except to GO to another spreadsheet or to execute another macro).
>
> If you want to have the macro stop at certain points to accept input from the operator (such as daily sales amounts, daily cash receipts, etc.), you must position the cursor at the location, enter the period (.) or comma (,) command, and then enter an up-arrow (^) immediately followed by a sample entry. Since the purpose here is just to "build" the macro, you can enter sample data (after the up-arrow (^)) in each location, which will be replaced by actual data during execution of the macro. The up-arrow (^) directs the macro to pause and accept operator input. When input is completed, the macro will continue executing. Remember, you enter an up-arrow immediately followed by your entry of data (no <CR> after the up-arrow).

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

If, during the building of the macro, you execute a "/FI" command, the macro will be saved with the spreadsheet, and thereafter, each time the macro is executed, the spreadsheet will be filed by the macro.

When you want the macro to stop recording, enter "/MACRO" again. The recording process will stop and you will be prompted:

DAILY will be saved (Y or N):

Enter a "Y" and the macro called DAILY will be saved for future execution. If an "N" is entered, the entire macro sequence will be erased and no macro will be saved.

In order to execute the macro, all that is necessary is to enter:

Command: =DAILY

The "=" followed by the macro name, entered at the Command: line, will initiate the macro. At this point, the macro will begin execution, pause for entry where specified, and then start up again.

You will also need to delete or perhaps overwrite an existing macro on occasion. If you enter the command "/MACRO" again, and enter the name of an existing macro, the following prompt will be displayed:

Macro name: DAILY R=run, D=delete, O=overwrite:

Here, you can enter "R" to execute the macro (although normally the "=macro.name" will be used to execute a macro). A "D" will cause the macro to be deleted. The entry of an "O" will allow overwriting of the macro, which has the same effect as deleting the macro and starting the recording of another one with the same name.

The /MACRO instruction is very powerful and can make your use of COMPU-SHEET much easier. If you have a command (such as a selected print command) that you execute over and over, you can set it up as a macro and execute the complete sequence with only one command.

If a macro is created with the same name as the spreadsheet, the macro will be automatically executed upon entering the spreadsheet.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - DETERMINING SPREADSHEET SIZE = /SIZE - -

COMPU-SHEET will allow for any number of columns or rows on a spreadsheet. However, there is a limit to the spreadsheet size that is related to the maximum record (or item) size allowed by the operating system. Some systems have a maximum record size of 32K bytes. Others have a 64K maximum, and yet others have no limit to the record size.

The data that is displayed on the screen and printed is stored in one item in the spreadsheet file (DATA). The formulas are stored in another (CALCS). If the spreadsheet reads information out of the database or reads other spreadsheets, the information read is stored in a third item (WORK). And finally, the information describing the number of columns, column width and justification, description, macros, etc. are stored in a fourth item (COLROW).

Each of these four items are limited by the maximum record size. That is, if any one of the four items reach the maximum size, the spreadsheet will abort with an error situation that cannot be recovered. There is also a possibility that if any two or more begin to approach within approximately 20 percent of the maximum record size, the spreadsheet may exceed work space and abort.

Now, because of the way information is stored within these items, it takes a rather large spreadsheet to reach the limits. However, it is always wise to file your spreadsheet periodically just to avoid the possibility of an abort due to reaching the limits or for any other reason.

The "/SIZE" command exists to inform you of the current size of your spreadsheet. By executing this command you will be able to keep tabs on any size problems before they occur.

Upon entry of the command there will be a short pause, and the sizes of the four items will be displayed. For example:

Data: 15,844 Calcs: 12,237 Work: 922 Colrow: 258

Review the size of each of the items and file the spreadsheet (with the "/FS" command) when there is a possibility of exceeding the limits.

EXAMPLES:

Command: /SIZE
Action: This command displays the size of the spreadsheet on the second status line.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - DISPLAYING THE INTERNAL FORMAT OF DATA = /IFORM - -

COMPU-SHEET stores all numeric data that is entered or calculated internally to four decimal place accuracy. This enables COMPU-SHEET to keep track of decimal places for you. Normally, you never need to be concerned with the internal representation of the data you see on the screen. However, there are times when this information could be of value. The "/IFORM" command toggles the data display on the second status line back and forth between the external and internal representation. The internal format is: adjusted to 4 decimal places and not masked (periods, commas...). External format is the same form as the data displayed within the spreadsheet, with the justification applied. There is no change in the data itself, only in the way it is displayed on the status line. Examples of external and internal formats are:

External format: 1,234.56

Internal format: 12345600

EXAMPLES:

Command: /IFORM
Action: If the data display on the second status line is displaying the data in external format, it will be changed to the internal format. If the data is being displayed in internal format, it will now be displayed in external format.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - DEFINING COLUMN WIDTH = /WIDTH - -

The "/WIDTH" command allows you to change the width of the locations within a column or range of columns. The width is the maximum number of characters which can be displayed for any location within a column.

Upon entry of "/WIDTH", the command line will request:

Column: enter the column number or range of column numbers to change.

To change a single column width, enter the column number (such as 8).

To change a range of column widths, enter the range in the format "col-col" (for example: 8-15). If a range of columns is entered, each column within the range will be set to the indicated width.

To change every other (or every third...) column width, enter the range in the format "col-col,x" (for example: 2-24,2) where "col-col" indicates the column range and "x" is optional and indicates the step (2=every other column, 3=every third column, etc...). If "x" is omitted, every column will be changed.

Column ranges can be stacked in groups such as: "2/5/7-11" where columns 2,5,7,8,9,10 and 11 would be changed with one command.

Upon entry of a single column number, the second status line will display the width and justification of the selected column.

The command line will then request:

Width: enter the desired width of the locations within the column or range of columns.

At this point, the column widths will be changed and the entire screen will be redisplayed.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

EXAMPLES:

Command: /WIDTH Column: 5 Width: 12
Action: This command will change each location within column 5 to a width of 12 and redisplay the screen.

Command: /WIDTH Column: 8-15/22-28,2 Width: 10
Action: This command will change each location within columns 8, 9, 10, 11, 12, 13, 14, and 15 to a width of 10, every other column from 22 thru 28 to a width of 10, and then will redisplay the screen.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - SETTING COLUMN JUSTIFICATION AND MASKING = /JUST - -

The "/JUST" command allows changing of the column justification and the mask applied to a column or range of columns. The "mask" indicates the number of decimal places, any commas, dollar signs, credit symbols, etc.

All numeric entries for columns with a mask are stored internally with 4 decimal places. The mask directs COMPU-SHEET to round (if necessary) and display the number of decimal places you have indicated.

Upon entry of the command, the command line will request:

Column: enter the number of the column or range of columns to change

To change a single column justification, enter the column number (such as 8).

To change a range of column justifications, enter the range in the format "col-col" (for example: 8-15). If a range of columns are entered, each column within the range will be set to the indicated justification and mask.

To change every other (or every third, etc.) column justification, enter the range in the format "col-col,x" (for example: 2-24,2) where "col-col" indicates the column range and "x" is optional and indicates the step (2=every other column, 3=every third column, etc...). If "x" is omitted, every column will be changed.

Column ranges can be stacked in groups such as: "2/5/7-11" where columns 2,5,7,8,9,10 and 11 would be changed with one command.

Upon entry of a single column number, the second status line will display the width and justification of the selected column.

Then you will be prompted:

Justification: enter the new justification for this column or range of columns

The format for the justification is:

J{n}{,}{$}{C}{;q}

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

where:

J is "L" for left justification or "R" for right justification.

n is a single numeric digit defining the number of decimal places you wish displayed. If n=0, only whole numbers will be displayed and the decimal point will be suppressed. THIS CHARACTER SHOULD BE ENTERED FOR ANY COLUMNS WHICH WILL HAVE ARITHMETIC CALCULATIONS PERFORMED (set to "0" if no decimal places are to be displayed).

, is an optional parameter for output which causes commas to be inserted between every thousands position of the value.

$ is an optional parameter for output which causes a dollar sign to be appended preceding the converted output value.

C/D/M/E/N is an optional parameter that is a credit indicator and can be one of the following:

C causes negative values to be followed by the letters CR

D causes positive values to be followed by the letters DB

M causes negative values to be followed by a minus sign (-).

E causes negative values to be enclosed inside angle brackets (< and >).

N causes the minus sign on negative numbers to be suppressed (absolute value).

;q is an optional parameter which directs COMPU-SHEET to append the character or characters "q" in front of each location in this column. Usually, "q" will be a blank and is used when the column on the left is right justified and the column on the right is left justified. The blank in "q" will separate the two columns for better readability. If the "q" is to be a blank, hit your space bar once after the ";". If you prefer the columns to be separated by an asterisk, enter a "*" after the ";", etc.

Example: Column 6 contains "1234500" and justification is "R2", column 7 contains "ABCD" and justification is "L" the display of columns 6&7 will be: "123.45ABCD"

Column 6 contains "1234500" and justification is "R2", column 7 contains "ABCD" and justification is "L;b" (b stands for a space, not the character "b"), the display of columns 6&7 will be: "123.45 ABCD"

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

Some examples of the use of the justification and masking command follow:

JUSTIFICATION AND MASK	INTERNAL VALUE	DISPLAYED VALUE
R2	123456775	12345.68
R2,	123456745	12,345.67
L2	123456700	12345.67
L2,	123456700	12,345.67
R2,$	123456700	$12,345.67
R2,E	-123456700	<12,345.67>
R0,	123456700	12,346
R2,C	-123456700	12,345.67CR
R2,C	123456700	12,345.67bb
R2,M	-123456700	12,345.67-
R2,M	123456700	12,345.67
L2,;b (b=blank)	123456700	b12,345.67

EXAMPLES:

Command: /JUST Column: 5 Justification: L
Action: This command will left justify all entries in column 5.

Command: /JUST Column: 5-15 Justification: R2,
Action: This command will change the justification of columns 5 thru 15 to right justification, and will mask any numeric entries with two decimal places, and will place commas between each thousands position.

Command: /JUST Column: 5/11-20,3 Justification: R0,C;b
Action: This command will change the justification of column 5 and every third column from 11 to 20 (or 11,14,17 and 20) to right justification, and will round each numeric entry to the nearest dollar, will place commas between each thousands position, will place a "CR" after any negative value (two spaces after any 0 or positive value), and will insert a space between column 4 and column 5.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - INSERTING COLUMNS AND ROWS = /INS - -

The "/INS" command provides a means to insert new columns and rows into an existing spreadsheet.

The "/INS" command provides an option for adjusting all formulas which would be affected by the insertion of columns or rows. If selected, all absolute references to locations within formulas will be adjusted to reflect the insertion. For example, if the formula "2.7+3.9" (which adds column 2 row 7 to column 3 row 9) exists on the spreadsheet, and a new column was inserted between column 2 and column 3, this formula would be changed to "2.7+4.9" (which adds column 2 row 7 to column 4 row 9) if the adjustment option were selected. All relative references (Pn or Fn) will NOT be changed to reflect the insertion. See the section on "DEFINING FORMULAS" for more discussion on the adjustment of formulas and relative references.

Upon entry of "/INS", the command line will request:

Insert col "C" or row "R": enter a "C" to insert columns, "R" to insert rows

If inserting columns, the following prompts will display:

Insert COLUMNS will display on the second status line

Adjust formulas (Y or N): enter "Y" to adjust formulas, "N" for no adjustment

Number of columns to insert: enter the number of columns to insert

Before col #: enter the column number where insertion is to begin (if you want to ADD columns to the right of an existing spreadsheet, this number must be the number of the column where addition is to begin)

Width: enter width of new column or columns

Just: enter justification of new column or columns

At this point, the columns will be inserted and the screen will be redisplayed. Any columns to the right of the inserted columns will shift right. If the adjustment option was selected, the program will scan through the formulas and make the adjustments reflecting the insertion. As the adjustment process is proceeding, the locations will be displayed on the lower right of the screen.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

If inserting rows, the following prompts will display:

Insert ROWS will display on the second status line

Adjust formulas (Y or N): enter "Y" to adjust formulas, "N" for no adjustment

Number of rows to insert: enter the number of rows to insert

Before row #: enter the row number where insertion is to begin

At this point, the rows will be inserted and the screen will be redisplayed. Any rows below the inserted rows will shift down. If the adjustment option was selected, the program will scan through the formulas and make the adjustments reflecting the insertion. As the adjustment process is proceeding, the locations will be displayed on the lower right of the screen.

EXAMPLES:

Command: /INS
Insert col "C" or row "R": C
Adjust formulas (Y or N): Y
Number of columns to insert: 2
Before col #: 8
Width: 10
Justification: RO
Action: This command will insert 2 columns between columns 7 and 8. The new columns will have a width of "10" and a justification of "RO". All formulas with absolute references to locations changed by the insertion will be adjusted accordingly.

Command: /INS
Insert col "C" or row "R": R
Adjust formulas (Y or N): N
Number of rows to insert: 1
Before row #: 12
Action: This command will insert one row between rows 11 and 12. Each location on the new row will assume the width and justification defined for the existing columns. Formulas will not be adjusted.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - DELETION OF COLUMNS AND ROWS = /DEL - -

The "/DEL" command provides for the deletion of columns or rows. This command will remove the specified column(s) or row(s) from the spreadsheet, and will "pull up" any remaining columns or rows to fill in the gap. (If you want to "erase" the data in a column or row, rather than "remove" the column or row, use the "/CLR" command.)

The "/DEL" command provides an option for adjusting all formulas which would be affected by the deletion of columns or rows. If selected, all absolute references to locations within formulas will be adjusted to reflect the deletion. For example, if the formula "2.7+4.9" (which adds column 2 row 7 to column 4 row 9) exists on the spreadsheet, and column 3 was deleted, this formula would be changed to "2.7+3.9" (which adds column 2 row 7 to column 3 row 9) if the adjustment option was selected. All relative references (Pn or Fn) will NOT be changed to reflect the deletion. See the section on "DEFINING FORMULAS" for more discussion on the adjustment of formulas and relative references.

IMPORTANT: The deletion process automatically deletes all existing windows and replaces them with one window, the size of your screen.

Upon entry of "/DEL", the command line will request:

Delete col "C" or row "R": enter a "C" to delete columns, "R" to delete rows

If deleting columns, the following prompts will display:

Delete COLUMNS will display on the second status line

Adjust formulas (Y or N): enter "Y" to adjust formulas, "N" for no adjustment

Number of columns to delete: enter the number of columns to delete

Start deleting with col #: enter the column number where deletion is to start

At this point, the columns will be deleted and the screen will be redisplayed. Any columns to the right of the deleted columns will shift left. If the adjustment option was selected, the program will scan through the formulas and make the adjustments reflecting the deletion. As the adjustment process is proceeding, the locations will be displayed on the lower right of the screen.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

If deleting rows, the following prompts will display:

Delete ROWS will display on the second status line

Adjust formulas (Y or N): enter "Y" to adjust formulas, "N" for no adjustment

Number of rows to delete: enter the number of rows to delete

Start deleting with row #: enter the row number where deletion is to begin

At this point, the rows will be deleted and the screen will be redisplayed. Any rows below the deleted rows will shift up. If the adjustment option was selected, the program will scan through the formulas and make the adjustments reflecting the deletion. As the adjustment process is proceeding, the locations will be displayed on the lower right of the screen.

EXAMPLES:

Command: /DEL
Delete col "C" or row "R": C
Adjust formulas (Y or N): Y
Number of columns to delete: 2
Start deleting with col #: 8
Action: This command will delete columns 8 and 9.
All formulas with absolute references to locations changed by the insertion will be adjusted accordingly.

Command: /DEL
Delete col "C" or row "R": R
Adjust formulas (Y or N): N
Number of rows to delete: 1
Start deleting with row #: 12
Action: This command will delete row 12. Formulas on the spreadsheet will not be adjusted.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - COPYING COLUMNS AND ROWS = /COPY - -

The "/COPY" command provides the ability to copy the data, formulas, or both from one column or row to one or more columns or rows. The copy function is column to column(s) or row to row(s).

The "/COPY" command provides for the following copy processes:

Copy a single location to another location.

Copy a range of locations within a column to another column.

Copy a range of locations within a column to a range of other columns (repeat a column to multiple columns).

Copy a range of locations within a row to another row.

Copy a range of locations within a row to a range of other rows (repeat a row to multiple rows).

In all cases, you can copy data, formulas, or both.

Upon entry of "/COPY", the command line will prompt:

Col or Row: Enter "C" to copy columns, "R" to copy rows

From: enter "from" range, starting location and ending location

To copy a single location, enter the location in the format "col.row".

To copy a range of locations, enter the range in the format "col.row-col.row", where the first location is the beginning point and the second location is the ending point.

To: enter the destination starting location or starting location range

If copying a single location, enter the destination location in the format "col.row".

If copying a column to another column or a row to another row, enter the destination starting location in the format "col.row", where "col.row" is the destination starting point. (There is no need to enter the ending location).

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

If copying a column to a range of other columns or a row to a range of other rows (repeating), enter the destination range in the format "col.row-col.row", where the range of locations represent the range of starting points for the copy.

If you would like to copy a column or row to every other column or row within a range, enter the range in the format "col.row-col.row,x" where the range of locations represents the range of starting points for the copy and "x" indicates the step value (2=every other, 3=every third, etc...).

D,F,B: enter a "D" to copy data only, "F" to copy formulas only, "B" to copy both data and formulas.

At this point, the copy will take place and the screen will be redisplayed.

If a format error is made during entry of the copy parameters, the "beep" will sound and the prompts will be repeated.

EXAMPLES:

Command: /COPY Col or Row: C From: 3.4 To: 4.4 D,F,B: F
Action: This example will copy the formula in location 3.4 to location 4.4.

Command: /COPY Col or Row: R From: 3.4-8.4 To: 3.6 D,F,B: D
Action: This example will copy the data in the 5 row locations from 3.4 thru 8.4 to 5 locations in row 6, starting with 3.6 and ending with 8.6.

Command: /COPY Col or Row: C From: 3.4-3.10 To: 4.4 D,F,B: B
Action: This example will copy the formulas in the 7 column locations from 3.4 thru 3.10 to 7 locations in column 4, starting with 4.4 and ending with 4.10.

Command: /COPY Col or Row: R From: 3.4-8.4 To: 3.5-3.10 D,F,B: B
Action: This example will copy both data and formulas in the row locations 3.4 thru 8.4 to the 5 rows starting with 3.5 thru 3.10.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - REPEATING A LOCATION = /REP - -

The "/REP" command provides for repeating a location on the spreadsheet across a row or down a column. The data, formulas, or both can be repeated or the reference to another formula can be repeated.

Upon the entry of "/REP", the command line will prompt:

Col or Row: enter "C" to repeat the location down a column, "R" to repeat it across a row.

From: enter the source location, in the format "col.row"

To: enter the destination location or range of locations

If repeating across a range of locations, enter the starting and ending locations in the format "col.row-col.row", where the first location is the starting point, and the second location is the ending point.

If you would like to repeat a location to every other location within a range, enter the range in the format "col.row-col.row,x" where the range of locations represents the range of the repeat and "x" indicates the step value (2=every other, 3=every third, etc...).

You can indicate multiple ranges for repeating by separating groups by a slash "/" such as: col.row-col.row,x/col.row-col.row,x.

D,F,B,$: enter a "D" to repeat data only, "F" to repeat formulas only, "B" to repeat both data and formulas, or a "$" to repeat the reference to the source location (see the section "REFERENCING OTHER FORMULAS").

At this point, the repeat will take place and the screen will be redisplayed.

If a format error is made during entry of the repeat parameters, the "beep" will sound and the prompts will be repeated.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

EXAMPLES:

Command: /REP Col or Row: R From: 3.8 To: 4.8-15.8 D,F,B: D
Action: This example will repeat the data in location 3.8 across row 8, starting with 4.8 and ending with 15.8.

Command: /REP Col or Row: C From: 3.8 To: 3.10-3.20,2 D,F,B: F
Action: This example repeats the formulas in location 3.8 to every other location in column 3 starting with 3.10 and ending with 3.20.

Command: /REP Col or Row: R From: 3.8 To: 4.8-15.8 D,F,B: $
Action: This example repeats the reference to the formula in location 3.8 ($3.8) across row 8 starting with column 4 and ending with column 15.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - MERGING SPREADSHEETS = /MERGE - -

The "/MERGE" command provides the means to retrieve all or part of other spreadsheets and either copy or add them into the current spreadsheet. This command is used to consolidate similar spreadsheets into a single document. Any block of locations from any spreadsheet within the current file can be merged. Formulas are copied if requested.

Upon entry of the "/MERGE" command the following prompts will occur:

Name: enter the name of the spreadsheet to be merged (it must exist in the same file as the current spreadsheet). If the named spreadsheet has a password that is different from the current spreadsheet the password must be entered. The format is: "name,password"

From: enter the locations to merge (from the source spreadsheet) in the format "col.row-col.row" where the first "col.row" is the upper left location to merge and the second "col.row" is the lower right location to merge

To: enter upper left location on the current spreadsheet where merging is to begin in the format "col.row"

D,F,B: enter a "D" to merge data only, "F" to copy formulas only, or "B" to merge data and copy formulas

Add: enter "Y" to add the numeric data in the source spreadsheet to the numeric data in the current spreadsheet, or "N" to copy the data in the source spreadsheet over any existing data in the current spreadsheet.

At this point, the merging process will occur and the screen will redisplay.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

EXAMPLES:

Command: /MERGE
Name: BUDGET1
From: 1.1-25.50
To: 1.1
D,F,B: B
Add: N
Action: This command will copy both data and formulas from the spreadsheet named "BUDGET1" in columns 1 thru 25, rows 1 thru 50, to the current spreadsheet, starting with column 1 row 1.

Command: /MERGE
Name: BUDGET2
From: 2.4-25.50
To: 2.4
D,F,B: D
Add: Y
Action: This command will add the data contained on the spreadsheet named "BUDGET2" in columns 2 thru 25, rows 4 thru 50, to the existing data in the current spreadsheet, starting with column 2 row 4. Only numeric data will be added.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - SORTING SPREADSHEET ROWS = /SORT - -

The "/SORT" command provides the ability to sort (numerically or alphabetically) the data on your spreadsheet by row. Rows can be sorted by the value in any column in either ascending or descending order.

After entering the command, you will be prompted:

Key col:

Enter the column number which is to be used as the sort key. A word of caution - from a sorting standpoint, spaces and blank locations have a lower value than other characters. You will then be prompted:

Start row:

Enter the row number where the sorting process is to begin. This is the first row of the range of rows being sorted. Note that each row is sorted in its entirety. You will then be prompted:

End row:

Enter the last row of the set of rows to be sorted. The next prompt is:

A/D(scending):

Enter an "A" to sort the rows in ascending sequence or a "D" to sort the rows in descending sequence (based upon the key column entered above). The last prompt is:

Formulas-Y/N:

The sorting process will sort the rows based upon the value of the key column. Each sorted row may end up in a different row than prior to the sort. The formulas associated with the row can be rearranged with the associated data row or the formulas can be left in their original position. Enter a "Y" to rearrange the formulas, or an "N" to leave them in their original positions.

At this point, the spreadsheet will be sorted and redisplayed.

IMPORTANT: It is recommended that you file (/FS) your spreadsheet PRIOR to executing the /SORT command. If the command is not entered correctly, it can scramble your spreadsheet. You can use the filed spreadsheet to recover from any unexpected results.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

EXAMPLES:

Command: /SORT
Key col: 2
Start row: 4
End row: 25
A/D(scending): D
Formulas-Y/N: N
Action: The spreadsheet will be sorted in descending order based upon the values in column 2. The range of rows being sorted begin with row 4 and end with row 25. The formulas will not be rearranged with the data.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - SORTING BLOCKS OF LOCATIONS = /SORTB - -

The "/SORTB" command (block sort) is used to sort any portion of your spreadsheet by either columns or rows. This is a more powerful command than the /SORT command, and is a little more difficult to use. However, it is useful when there is a need to sort spreadsheet columns or a portion of a spreadsheet.

After entering the command, you will be prompted:

Cols/Rows:

Here, you need to indicate whether you want to sort spreadsheet columns or rows. Enter a "C" to sort columns or "R" to sort rows. The next prompt is:

Range:

Enter the range or block of locations to sort. Enter the upper left location and the lower right cell of the block in the format "col.row-col.row". This block outlines the section of cells which will be sorted. The next prompt is:

Key Col: or Key Row:

Here, you must indicate the column or row which contains the sort key. If you are sorting columns, this will be the row which contains the values to sort by. If you are sorting rows, this will be the column which contains the values to sort by. A word of caution - from a sorting standpoint, spaces and blank locations have a lower value than other characters. The next prompt:

A/D(scending):

Indicate the order you wish to sort. You have the option to sort the columns or rows in either ascending or descending sequence. Enter an "A" to sort in ascending sequence or a "D" to sort in descending sequence. The last prompt is:

Formulas:

If you want to rearrange the formulas according to the result of the sort, enter a "Y". If you do not want to rearrange the formulas, enter an "N".

At this point, the spreadsheet will be sorted and redisplayed.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

IMPORTANT: It is recommended that you file (/FS) your spreadsheet PRIOR to executing the /SORTB command. If the command is not entered correctly, it can scramble your spreadsheet. You can use the filed spreadsheet to recover from any unexpected results.

EXAMPLES:

Command: /SORTB
Cols/Rows: C
Range: 2.5-10.25
Key Row: 25
A/D(scending): A
Formulas: N
Action: This example shows how to sort a spreadsheet by columns. The block being sorted is columns 2 thru 10 and rows 5 thru 25. The sort will be in ascending sequence based upon the values in row 25. Formulas will not be rearranged.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - WINDOW MANIPULATION COMMANDS - -

The window commands provide the means to split the screen into groups of one or more columns. Any window can scroll in any direction over the screen. For example, this gives the user the ability to keep a description column on the screen while scrolling around the rest of the spreadsheet.

IT IS IMPORTANT TO NOTE THAT ALL WINDOWS WILL BE DISPLAYED AT ONE TIME. This means the combined width of all windows must not be larger than the width of your screen.

The "/WINDOW", "/SEL", and "/SETH" commands are the three commands that will be used most frequently in defining and working with windows. The "/WINDOW" is used to define one or more vertical windows on your screen, and the "/SEL" command is used to select one of the defined windows for placement of the cursor. The "/SETH" command is used to "lock" a set of column headings on the screen.

When more than one vertical window is defined, you will find that the window cursor will move within only one of the windows. This window is called the "selected window" and can be changed with the "/SEL" command. The selected window will scroll from column to column, in either direction. As the left or right margin of the selected window is "bumped", that window will scroll to the left or right accordingly. Any other windows will remain as they were. As the top or bottom margin of the selected window is "bumped", ALL windows will scroll up or down together so they remain synchronized. If you move the cursor to a column that is already being displayed in another window, the column will be displayed twice - once in the selected window, and again in the other window.

If multiple windows are defined, the "page" mode will automatically be set to "N" (off).

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - DEFINING VERTICAL WINDOWS = /WINDOW - -

The "/WINDOW" command provides an easy way to split your screen into vertical windows. With this command you can display different sets of columns on the screen at one time.

Each window can display one or more columns from different parts of your spreadsheet. You can, for example, display column 1 on the left side, columns 5-10 in the center, and column 25 on the right side...all at the same time.

Upon entry of the command, you will be prompted:

Enter window settings:

Enter the window settings in the format:

col-range/col-range/col-range....

Where a "col-range" can be either one column number (such as: 1), or a range of column numbers (such as: 5-10).

If the requested window definition exceeds the screen size, an error message will appear and you will need tc redefine the windows to consist of fewer columns.

The window selected for placement of the cursor will be:

window 1 - if only 1 window is defined

window 2 - if 2 or more windows are defined

EXAMPLES:

Command: /WINDOW
Enter window settings: 1-8
Action: This command will remove all previously defined windows and will create one window, consisting of columns 1 thru 8.

Command: /WINDOW
Enter window settings: 1/5-10/25
Action: This command will define 3 windows. Window 1 will consist of only column 1, window 2 will consist of columns 5 thru 10, and window 3 will consist of column 25. The cursor will be placed in window 2. You will be able to scroll window 2 over the entire spreadsheet and window 1 (column 1) and window 3 (column 25) will remain on the screen.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - SELECTING A WINDOW = /SEL - -

The "/SEL" command selects a window. The cursor will only scroll within one window. Therefore, if the user wishes to access locations in another window, the "/SEL" command must be executed to move the cursor to the desired window.

Upon entry of "/SEL", the command line will request:

WINDOW #: enter the window number you wish to select.

The window requested will be selected. The "window cursor" will move to the window and the status lines will reflect the selected window.

EXAMPLES:

COMMAND: /SEL WINDOW #: 2
Action: This example selects window 2 (previously defined). The window cursor will be located in the upper left location of window 2 and can be moved in any direction.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - DEFINING OR DELETING A SINGLE WINDOW = /SET - -

The "/SET" command provides for defining or deleting a window. It will allow you to define only one window at a time. It will allow you to define a window without requiring rows to be aligned with other windows.

This command is useful, for example, when you want two windows displayed where the first window consists of columns 1, 2, 3 and rows 1 thru 20. The second window may consist of the same columns, and rows 21 thru 40. In this example, we do not have row alignment.

Upon entry of the "/SET" command you will be prompted:

Enter window number: enter the window number to be set

Window starting location (col.row) or "DELETE": enter the starting location (upper left) of the window

> The starting location is entered in the format "col.row", where "col" is the starting column and "row" is the starting row. This location will be the upper left location of the window.
>
> If "DELETE" is entered, this window will be deleted and any windows with a number greater than the one deleted will be shifted to the left to fill the deleted window number.

End column (col): enter the ending column number (upper right)

> Here, enter the number of the rightmost column of the window. If the window is only one column wide, then this column number will be the same as "col" in the previous prompt. Note the row number is not entered.

Suppress row numbers (Y or N): enter a "Y" to suppress the row numbers, "N" to display the row numbers

> If you wish to suppress the display of the row numbers associated with each row of this window, enter a "Y". If you wish to display the row numbers, enter a "N".

At this point, the screen will be redisplayed. All defined windows will be included in the display.

If a format error is made during the entry of the parameters, a "beep" will sound and the prompts will be repeated.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

EXAMPLES:

Command: /SET
Enter window number: 1
Window starting location (col.row) or "DELETE": 1.1
Window ending column (col): 3
Suppress row numbers (Y or N): N
Action: This command defines window 1 as consisting of columns 1 thru 3 and rows 1 thru 20. The row numbers will be displayed.

Command: /SET
Enter window number: 2
Window starting location (col.row) or "DELETE": 1.21
Window ending column (col): 3
Suppress row numbers (Y or N): N
Action: This example defines window 2 as consisting of columns 1 thru 3 and rows 21 thru 40. These two examples show how you can split the screen into windows which, in effect, "stack" 40 rows on one screen.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - LOCKING COLUMN HEADINGS = /SETH - -

The "/SETH" command provides the means to "lock" a number of "column heading" rows on the screen. This feature aids in determining column identification as the windows move down the spreadsheet.

Upon entry of "/SETH", the command line will prompt:

Heading row #: enter the number of rows to lock as headings

Rows 1 thru the number entered here will be "locked" on the screen.

If no entry is made (a "CARRIAGE RETURN" only), any prior /SETH entry will be cleared.

At this point the screen will be redisplayed with "heading rows" locked or unlocked as indicated. When the cursor is moved downward, beyond row 20, the heading rows 1 thru the number entered will remain on the screen.

EXAMPLES:

Command: /SETH Heading row #: 5
Action: This example will "lock" rows 1 thru 5 on the screen as column headings. When the cursor is moved below row 20, rows 1 thru 5 will remain on the screen.

Command: /SETH Heading row #: <CR>
Action: This example will clear any previously "locked" headings.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - SUPPRESS ROW NUMBERS = /SUPP - -

The "/SUPP" command will allow for suppression of the row numbers associated with the currently selected window.

Upon entry of "/SUPP", the command line will prompt:

Suppress row number? (Y or N or <CR>=N):

Enter a "Y" to suppress the display of the row numbers or a "N" to cause the row numbers to display.

At this point, the screen will be redisplayed.

EXAMPLES:

Command: /SUPP
Suppress row numbers? (Y or N or <CR>=N): Y
Action: This example suppresses the display of the row numbers of the current window. The spreadsheet will be redisplayed.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - STORING OR DELETING THE SPREADSHEET = /FILE or /FI or /FS - -

These commands provide for storing (saving) the spreadsheet in the file, or deleting the spreadsheet from the file. The spreadsheet name or password can be changed, or the entire spreadsheet can be deleted with these commands.

The "/FILE" and "/FI" commands operate alike. They will store or delete the spreadsheet and return to the beginning of the program at the file name prompt.

The "/FS" command stores the spreadsheet and then returns to the "Command:" prompt instead of exiting the spreadsheet.

Upon entry of any of these commands, the status lines will be replaced with:

Write the spreadsheet: (name) on the file: (name)
Enter "Y" if OK...or enter a new spreadsheet name...or "DELETE" to delete it

Response:

Enter a "Y" to write the spreadsheet to the file under the name and password initially assigned.

Or enter a new spreadsheet name and password, if desired. The format is:

"name{,password}", where "name" is the new spreadsheet name (or the same name if changing only the password), and "password" is the new password (optional, and if omitted the password will be null). At this point, the prompt will be repeated with the new spreadsheet name.

(You still have the same spreadsheet stored under the old name "without" the recent updates, and the spreadsheet stored under the new name "with" the recent updates. This can be helpful when using the same format for several different spreadsheets. Create the master format and change its name each time you wish to create an identical format to input new information.)

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

If a COMPU-SHEET spreadsheet was previously stored under this name, the following warning message will display:

A spreadsheet already exists with this name - overwrite it? (Y or N):

Enter a "Y" to overwrite the previous spreadsheet, or an "N" to return to the first prompt.

Enter the word "DELETE" to delete this spreadsheet from the file.

It is recommended that spreadsheet files are periodically reviewed for obsolete spreadsheets, which should be deleted. By purging your files of obsolete spreadsheets, you can save valuable disk space on your system and speed up the processing of the remaining spreadsheets in the file.

At this point, the command will be executed and COMPU-SHEET will return to the "Welcome to COMPU-SHEET" prompt (for /FILE and /FI), or will return to the "Command:" prompt (for /FS).

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - DEFINING PRINTER AND SCREEN SIZES = /TERM - -

The "/TERM" command provides for setting the printer width in characters and printer depth in lines, and the screen width in characters. These parameters control the point where printed output will "fold" (paginate based upon line width exceeding printer width) and "overflow" (paginate based upon lines printed exceeding the number of lines per page). It also will determine the size of the spreadsheet display on your terminal.

Upon entry of the command, you will be prompted:

Printer width in characters: enter the number of characters to print on each line

Printer depth in lines: enter the number of lines to print per page

Screen width in characters: enter the width of your screen in characters (normally either 80 or 132).

This command overrides the printer width and depth defined in the terminal definition process for this port.

EXAMPLES:

COMMAND: /TERM
Printer width in characters: 136
Printer depth in lines: 80
Screen width in characters: 132
Action: This command defines a printer width of 136 print positions, a printer depth of 80 lines, and a screen width of 132 characters.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - PRINTING A SPREADSHEET = /PRINT and /PRINTT - -

The "/PRINT" command will provide for printing all or part of the spreadsheet on the line printer.

The "/PRINTT" command will provide for printing all or part of the spreadsheet on a terminal printer.

These print commands offer flexibility in printing your spreadsheet.

Upon entry of the "/PRINT" or "PRINTT" command, you will be prompted:

Print all or select columns and rows (An or Sn):

Enter an "A" to print the entire spreadsheet. If the size of the spreadsheet exceeds the printer width or depth, the printed spreadsheet will fold and/or overflow according to the defined printer width and depth.

Enter an "S" to select the columns and rows to print, and/or to repeat certain columns and/or rows on each printed page.

The "n" following the "A" or "S" indicates the number of copies you would like to print. If "n" is omitted, 1 copy is printed. For example, "A3" indicates to print 3 copies of the entire spreadsheet.

If "A" for "all" was selected, the following prompts will display:

Print column and row numbers (Y or N, <CR>=N): enter a "Y" to print column and row numbers on the spreadsheet. Enter a "N" or <CR> to suppress the printing of the column and row numbers on the printed spreadsheet.

At this point the entire spreadsheet will be printed according to the format of the command.

If "S" for "select" was entered, the following prompts will display:

Repeat cols "C", rows "R", both "B", or <CR>: enter a "C" to repeat the first group of columns selected on any page folds, "R" to repeat the first group of rows selected on page overflow, "B" to repeat the first group of columns and rows on page folds and overflow. A <CR> will not repeat either columns or rows on page folds or overflow.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

Columns to print: enter the list of columns you wish to print in the format "n1/n2-n3/n4/n5-..." where n1 represents the first column or group of columns and will repeat on page folds if you had requested it, "n2-n3" represents any range of columns, "n4" and "n5-..." represents any combination of columns, etc... If a semi-colon ";" is used to separate a group of columns instead of a slash (/), a page break will occur between the columns separated by the semi-colon (;).

Rows to print: enter the list of rows you wish to print in the format "n1/n2-n3/n4/n5-..." where n1 represents the first row or group of rows and will repeat on each page overflow if you had requested it, "n2-n3" represents any range of rows, "n4" and "n5-..." represents any combination of rows, etc... If a semi-colon ";" is used to separate a group of rows instead of a slash (/), a page break will occur between the rows separated by the semi-colon (;).

Print column and row numbers (Y or N or <CR>=N): enter a "Y" to print column and row numbers on the printed spreadsheet, "N" or <CR> to suppress the column and row numbers.

At this point, the selected columns and rows will be printed and COMPU-SHEET will return to the command prompt.

If a format error is made during the entry of the parameters, a "beep" will sound and the prompts will be repeated.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

EXAMPLES:

Command: /PRINT
Print all or select columns or rows (An or Sn): A2
Print column and row numbers (Y or N, <CR>=N): N
Action: This command will print the entire spreadsheet wihout the column and row numbers on the line printer. Two copies will be printed.

Command: /PRINT
Print all or select columns or rows (An or Sn): S
Repeat cols "C", rows "R", or both "B", or <CR>: B
Enter the columns to print: 1/2-8/15-21
Enter the rows to print: 1-3/4-120
Print column and row numbers (Y or N, <CR>=N): Y
Action: This command will print a selected list of columns and rows on the line printer. Columns 1 thru 8, and 15 thru 21, and rows 1 thru 120 will print. Column 1 will repeat on each page fold, and rows 1 thru 3 will repeat on each page overflow. The column and row numbers will print on each page. One copy will be printed.

Command: /PRINTT
Print all or select columns or rows (An or Sn): S3
Repeat cols "C", rows "R", or both "B", or <CR>: <CR>
Enter the columns to print: 1/8-12;19-28
Enter the rows to print: 1-45
Print column and row numbers (Y or N, <CR>=N): N
Action: This command will print columns 1 and 8 thru 12 on the first page and columns 19 thru 28 on the second page. Rows 1 thru 45 will be printed on the terminal printer. The column and row numbers will not print and no columns or rows will be repeated on each page overflow. Three copies will be printed.

Command: /PRINT
Print all or select columns or rows (An or Sn): S
Repeat cols "C", rows "R", or both "B", or <CR>: C
Enter the columns to print: 1/2-36
Enter the rows to print: 1-45
Print column and row numbers (Y or N, <CR>=N): N
Action: This command will print spreadsheet columns 1 thru 36 and rows 1 thru 45 on the line printer. Column 1 will repeat on every page and COMPU-SHEET will determine where the page breaks will occur. One copy will be printed.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - AUDITING A SPREADSHEET = /AUDIT and /AUDITT - -

The "/AUDIT" command provides for printing an audit list of all or part of a spreadsheet on the line printer.

The "/AUDITT" command provides for printing an audit list of all or part of a spreadsheet on a terminal printer.

Both commands will provide automatic page folding and overflow according to the defined printer width and depth.

Both commands will print a formatted audit listing containing the column widths and justifications for the select range of locations. Each selected location will display the formatted data, unformatted data, and any formula associated with the location. Also, the spreadsheet description, date and time of last update, operator initials, and other information will be printed.

Upon entering either command, you will be prompted:

Enter locations (col.row-col.row) or 'ALL':

> Enter either "ALL" or a range of locations to audit in the format "col.row-col.row", where the first location is the upper left location to audit and the second location is the lower right location to audit. All locations within this range will be printed.

At this point, an audit report of the selected columns and rows will be printed and COMPU-SHEET will return to the command prompt.

If a format error is made during entry of the parameters, a "beep" will sound and the prompts will be repeated.

EXAMPLES:

Command: /AUDIT
Enter locations (col.row-col.row) or 'ALL': 3.5-10.5
Action: This example will print an audit report on the line printer of row 5, starting with column 3 and ending with column 10.

Command: /AUDITT
Enter locations (col.row-col.row) or 'ALL': ALL
Action: This command will print an audit report of the entire spreadsheet on the terminal printer.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - CREATING A FILE FROM THE SPREADSHEET = /FILBLD - -

COMPU-SHEET provides the user with the ability to output a spreadsheet to a data file for processing by other applications. You have the ability to create the data level or both the dictionary and data levels. This means that a spreadsheet can be turned into a standard file which can be processed by "ACCESS", "ACCU-PLOT", or any other standard processor or utility.

The "/FILBLD" command creates a standard file from the contents of the spreadsheet. You will specify the columns and rows to output. Each specified row is written as an item in the data file with each specified column corresponding to an attribute in the item.

Upon entering the command, you will be prompted:

Enter the name of the file which is to receive the spreadsheet file:

> Enter the name of the data file which is to be written by this process. The specified file must be authorized to receive output from this process. The dictionary of the file must contain an item with an ID of "CS%FILBLD". This item must have an "X" in attribute 1. If attribute 2 to "n" contains spreadsheet names, only the spreadsheets indicated will be allowed to create the file. If attribute 2 is null, any spreadsheet can write to the file. This procedure will prevent a user from accidentally writing to a file which was not intended to be used with this process.

Heading rows:

> If you want the process to build the dictionary, enter the row numbers which contain the column headings to be inserted into the dictionary items. These headings will appear as column headings when the file is processed by ACCESS. If you do not want to build the dictionary, enter "<CR>". The format is "row/row-row/row-row....".

Cols:

> Enter a list of columns which are to be output as attributes in the generated items. The columns will be assigned attribute numbers in the order of specification. The format is "col/col-col/col-col....".

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

Rows:

Enter a list of rows which are to be output as items in the file. Each item will be assigned an item-ID consisting of its associated row number. The format is "row/row-row/row-row....".

At this point, the spreadsheet will be written to the file. The data will be written as it appears on the spreadsheet. That is, the data written to the file will contain the same number of decimal places as is displayed in the column.

EXAMPLES:

Command: /FILBLD
File: BUDGETS
Heading rows: 3-4
Cols: 1-10/16
Rows: 6-33
Action: This command will write this spreadsheet to the file called "BUDGETS". The dictionary will be built and the headings to place in the dictionary are in rows 3 and 4. Each item will have 11 attributes numbered 1 to 11 and will consist of the contents of columns 1 thru 10 and 16. There will be 27 items created, one for each row from 6 thru 33. The item-ID will be the same as the row numbers.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - EXITING THE COMPU-SHEET SPREADSHEET = /X - -

The "/X" command will cause COMPU-SHEET to exit from the current display and return to the "WELCOME TO COMPU-SHEET" prompt. Any changes to the item will NOT be stored.

As a precaution, if the "/X" command is entered, a warning message is printed:

WARNING - updates will NOT be saved - enter "Y" to exit:

Enter a "Y" to exit the spreadsheet. Any other response will return to the "Command:" prompt.

EXAMPLES:

Command: /X
WARNING - updates will NOT be saved - enter 'Y' to exit: Y
Action: This example will cause an exit from COMPU-SHEET. Any changes to the spreadsheet item since it was last filed will be lost.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - DEFINING FORMULAS - -

COMPU-SHEET offers you the ability to compute the value of any location by performing mathematical and logical operations on one or more locations, constants, or data retrieved from other files or spreadsheets in your system. If you can program, you can write your own subroutines which can be called from any formula. These subroutines can manipulate the spreadsheet as needed and then pass the modified spreadsheet back to COMPU-SHEET.

The simplicity, flexibility, and power of these calculations gives you the potential to solve even the most complex business problems.

FORMULA FORMATS

There are two types of formula formats recognized by COMPU-SHEET. The first is the algebraic notation format. There is no need to be concerned about your math skills. If you can do simple addition, subtraction, multiplication and division, you can write formulas for COMPU-SHEET. The second format utilizes a "stack" processor (sometimes called the "RPN" or "Reverse Polish Notation" format). This format is provided for those users who prefer it over the algebraic form; however, most users will find the algebraic format easier to use.

The following discussions first cover the algebraic format in detail, and then the concepts and operation of the stack processor.

FORMULA ENTRY

All formulas are entered at the FORMULA: prompt. The prompt is displayed as part of the data/formula entry sequence and is initiated by the period (.) and comma (,) commands. Upon entry of one of these commands, you will be prompted for data entry into the window cursor in the spreadsheet (unless a formula had previously been entered, in which case you will be prompted for FORMULA:). If no data is entered (a <CR> at the request for data entry), you will be prompted for FORMULA:. At that point, you can enter any formula in either format.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

ERROR MESSAGES

When a formula is entered, COMPU-SHEET evaluates the formula to determine if it was entered correctly. It checks the formula for proper format, makes sure you don't attempt to perform arithmetic calculations on locations that contain alphabetic information, makes sure you don't attempt an invalid operation such as division by zero, and a number of other checks.

If upon evaluation, COMPU-SHEET determines the formula is incorrect, an error message will be displayed in the location containing the formula that is incorrect. This error message appears as follows:

ERRn

where "ERR" indicates the formula in this location is invalid, and "n" is a number which indicates the type of error encountered.

A list of error message numbers follow. Some of the error messages described below will be displayed by the algebraic format processor and others will be displayed by the function format processor.

ERR #	DESCRIPTION
ERR1	The formula has been written incorrectly. There may be an error in the format, a missing comma, quote, parenthesis, etc.
ERR2	The formula contains an invalid spreadsheet location reference.
ERR3	An arithmetic operation was attempted on non-numeric data. One or more locations or literals contain spaces or other alphabetic characters.
ERR4	An attempt was made to divide by zero or some other operation was attempted where zero is an illegal operand.
ERR5	An invalid "jump" command format was found (function format only).
ERR6	An attempt was made to raise a negative number to a non-integer power.
ERR7	An operation attempting to sum a multi-valued attribute has encountered non-numeric data.
ERR8	The formula contains an invalid text extract format.
ERR9	The file specified in a "READ" operation cannot be located in this account.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

ERR10 An attempt has been made to read or write a file in the database without authorization to access the database.

ERR11 An attempt has been made to read a file in the database without authorization to read the database.

ERR12 An attempt has been made to write to a file in the database without authorization to write to the database (function format only).

ERR13 The item specified in a "READ" operation cannot be located in the specified file.

ERR14 There is an invalid format in a "goto" (function format only).

ERR15 The "goto" function will be executed by the "/CAL" command (function format only).

ERR16 The "FIND" function format is invalid.

ERR17 The value to "find" is less than the first entry in the table.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - THE ALGEBRAIC FORMAT - -

The algebraic format is simple to write and easy to understand. The general form of the algebraic format is:

expression

Where "expression" is made up of one or more elements which define your formula. The result of this formula is always stored and displayed in the location containing the formula. Let's preview a couple of formulas just to show you the basic concept:

2.5+2.6 (which adds the contents of locations 2.5 and 2.6, and displays the sum in the location containing this formula)

2.5*(2.6+2.7) (which multiplies the contents of location 2.5 by the sum of the contents of locations 2.6 and 2.7, and displays the result in the location containing the formula)

Of course these are simple examples, but they serve to demonstrate the basic concept of algebraic formulas. The following discussions cover the concepts you need to know to build algebraic formulas.

BASIC ARITHMETIC OPERATORS

The operators + - * and / are available and take two operands (spreadsheet locations, literals, results of special functions...) and return their sum, difference, product, or quotient. It is important to note that all arithmetic calculations are carried to 4 decimal positions. The arithmetic processor will maintain the decimal location for you in all calculations.

The multiplication (*) and division (/) operators will be executed before the operators for addition (+) and subtraction (-). You can use parentheses to override the normal order of operations. For example, if a location contained the formula: 2.5*2.6+2.7, it would first multiply the contents of locations 2.5 and 2.6 and then add the product to the contents of location 2.7. However, if the formula was written as 2.5*(2.6+2.7), the contents of locations 2.6 and 2.7 would be added together and then their sum would be multiplied by the contents of location 2.5. The parentheses have overridden the normal order of operations.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

SPREADSHEET LOCATIONS

A spreadsheet location is specified by placing the intersecting column and row numbers as an element in the expression. The format of the spreadsheet location is:

col.row

where "col" is any column number and "row" is any row number. A period separates the two numbers. An example of this format is:

2.5+2.6

which adds the contents of location 2.5 to the contents of location 2.6 - the result becomes the contents of the location containing the formula.

There are some variations to this method of referring to spreadsheet locations. The first involves the omission of the reference to the row number. The format is:

col.

where "col" is any column number, followed by a period. Note the row number is omitted. It is filled in at calculation time with the row number of the location containing the formula. For example, if location 5.10 contains the formula:

3.+4.

the row number "10" will be filled in at calculation time (thereby making the formula: 3.10+4.10). If the formula was in location 5.11, the row number "11" would be filled in at calculation time (thereby making the formula: 3.11+4.11).

The second variation involves the omission of the reference to the column number. The format is:

.row

where "row" is any row number, preceded by a period. Note the column number is omitted. It is filled in at calculation time with the column number of the location containing the formula. For example, if location 5.10 contains the formula:

.8+.9

the column number "5" will be filled in at calculation time (thereby making the formula: 5.8+5.9). If the formula was in location 6.10,

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

the column number "6" would be filled in at calculation time (thereby making the formula: 6.8+6.9).

There is a good reason why you might want to write formulas where the column or row numbers are omitted - to save you time and effort. You can see that the formula .8+.9 would operate on rows 8 and 9 of whatever column the formula happens to be in. Therefore, you can write the formula once and then use the "/REP" (repeat) command to repeat the formula to a range of other locations - saving you time.

There is still another method of referencing locations. This method is called "relative" referencing. In this format, instead of indicating the specific (absolute) column or row number, you can indicate that the column or row is a certain number of columns or rows away from the location containing the formula. In the format "col.row", either "col" or "row" may be in the format:

Pn - where "P" indicates "previous" columns or rows, and "n" indicates the number of previous columns or rows.

Or may be in the format:

Fn - where "F" indicates "following" columns or rows, and "n" indicates the number of following columns or rows.

Here are some examples which explain this concept further:

col.row	
5.10	- means "take the contents of the location at the intersection of column 5 and row 10".
P1.	- means "take the location in the previous column within the current row".
P2.5	- means "take the location at the second previous column on row 5". This example combines relative referencing for the column with absolute referencing for the row.
.P3	- means "take the location in the current column, three rows up (previous)".
5.P1	- means "take the location in column 5, one row up (previous)".
F1.3	- means "take the location in the following column, on row 3".
2.F5	- means "take the location in column 2, 5 rows down (following)".
P1.F2	- means "take the location in the previous column, 2 rows down (following)".

These various methods of referencing the contents of spreadsheet locations are used in building your formulas. You should become familiar with each of these formats so you can choose the most effective one in any given formula.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

LITERALS OR CONSTANTS

A "literal" or "constant" is any numeric value or series of alphanumeric characters. The terms "literal" and "constant" are interchangeable. A literal can be a number such as "123.45" or ".005" or "-100.00" or it can be a series (string) of characters such as: "ABCD" or "AMOUNT" or "A1" or "******". Literals used in a formula must have double quote signs surrounding the value being defined.

Literals are used in formulas to indicate "use this specific value".

For example, the formula:

4.5*"1.5"

indicates the contents of spreadsheet location 4.5 is to be multiplied by the (literal) value "1.5". The value "1.5" is not a spreadsheet location, it is the actual value "1.5".

Any literal string or numeric constant can be specified by enclosing the value in double quotes. If the literal is a numeric value that is to be used in an arithmetic operation, THE DECIMAL POINT MUST BE INDICATED, EVEN IF THE NUMBER IS A WHOLE NUMBER. For example, if the whole number "two" is used as a literal in an arithmetic operation, the decimal point must be entered:

4.5*"2."

This indicates the contents of spreadsheet location 4.5 is to be multiplied by the (literal) value "2". The decimal point must be indicated in the literal, even when there are no decimal places.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

SPECIAL FUNCTIONS

There are several special functions which provide "short cuts" to performing common types of calculations. For example, a special function is available which provides for returning the total of a range of spreadsheet locations. Another one will calculate the average value of a range of spreadsheet locations. The following special operands must be surrounded with single quotes.

ACCUMULATE (SUM)

'A,loc1,loc2' - which will return the accumulated values
'A,loc1,loc2,n' of "loc1" thru "loc2" ("loc1" is the upper
'A+,loc1,loc2,n' left location and "loc2" is the lower right
'A-,loc1,loc2,n' location - all numeric values within these locations will be accumulated).
"n" may be used to indicate the occurance of the accumulate - such as: 2=every other location, 3=every third location...
The "+" or "-" indicate that only the positive or negative values should be accumulated.

Examples: 'A,2.5,2.35' - returns the sum of all numeric values in the locations in column 2, from row 5 thru row 35

'A,2.,24.,2'/26.3 - returns the sum of the locations in every other column, from column 2 thru column 24 (on the row containing the formula) divided by the contents of location 26.3

'A,.5,.P2' - returns the accumulated total of all numeric values in the locations in the current column, starting with row 5 thru 2 rows prior to the row containing the formula

AVERAGE

'AVG,loc1,loc2' - which will return the average value
'AVG,loc1,loc2,S' of "loc1" thru "loc2" ("loc1" is the
'AVG,loc1,loc2,n' upper left location and "loc2" is the
'AVG,loc1,loc2,S,n' lower right location). "S" is optional and indicates all zero values are to be excluded from the average. All numeric values will be averaged.
"n" may be used to indicate the occurance of the average - such as: 2=every other location, 3=every third location...

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

Examples: 'AVG,2.5,2.35' - returns the average of all numeric values in the locations in column 2, from row 5 thru row 35

'AVG,2.,24.,2' - returns the average of the numeric values in every other column, from column 2 thru column 24 (on the row containing the formula)

'AVG,.5,.P2,S' - returns the average of all non-zero values in the locations in the current column, starting with row 5 thru 2 rows prior to the row containing the formula

COUNT

'CNT,loc1,loc2' - which will return the count of all
'CNT,loc1,loc2,S' numeric values in the range from
'CNT,loc1,loc2,n' "loc1" thru "loc2" ("loc1" is the
'CNT,loc1,loc2,S,n' upper left location and "loc2" is the lower right location). "S" is optional and indicates all zero values are to be excluded from the count.
"n" may be used to indicate the occurance of the count - such as: 2=every other location, 3=every third location...

Examples: 'CNT,2.5,2.35' - returns a count of the number of numeric values in the locations in column 2, from row 5 thru row 35

'CNT,2.,24.,S,3' - returns a count of non-zero values in every third location in columns 2 thru 24 (on the row containing the formula)

MINIMUM

'MIN,loc1,loc2' - which will return the minimum value
'MIN,loc1,loc2,S' in the range from "loc1" thru "loc2"
'MIN,loc1,loc2,n' ("loc1" is the upper left location
'MIN,loc1,loc2,S,n' and "loc2" is the lower right location). "S" is optional and indicates zero values will not be returned. Only numeric values are considered.
"n" may be used to indicate the occurance of the minimum - such as: 2=every other location, 3=every third location...

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

Examples: 'MIN,2.5,2.35' - returns the minimum value in the range of locations in column 2, from row 5 thru row 35

MAXIMUM

'MAX,loc1,loc2' - which will return the maximum value
'MAX,loc1,loc2,S' in the range from "loc1" thru "loc2"
'MAX,loc1,loc2,n' ("loc1" is the upper left location
'MAX,loc1,loc2,S,n' and "loc2" is the lower right location). "S" is optional and indicates any zero value will not be returned. Only numeric values are considered.
"n" may be used to indicate the occurance of the maximum - such as: 2=every other location, 3=every third location...

Examples: 'MAX,2.5,2.35' - returns the maximum value in the range of locations in column 2, from row 5 thru row 35

OTHER SPECIAL FUNCTIONS

There are several more special functions which perform miscellaneous functions. The formats are different than those above. Note: these functions are not surrounded by single quote signs.

TEXT EXTRACT (SUBSTRING)

expression[n,m] - which returns a substring starting with character "n" for "m" characters from the resolved "expression". ("expression", "n" and "m" may be either a location or a literal)

Examples: 2.5["5","6"] - returns 6 characters from the contents of location 2.5, starting with the fifth character thru the tenth character

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

CONVERSIONS

(expression)(conversion) - which will perform the "conversion"
(expression)(I,conversion) on the resolved "expression".
If the (I,conversion) format is used, an "input" conversion is applied, otherwise an "output" conversion is applied. Conversions are special operating system routines which manipulate data according to the conversion process. For more information on conversions, please refer to your system manuals.

Examples: (3.5*3.6)(MR14) - returns the result of the contents of locations 3.5 times 3.6 rounded to one decimal place (instead of 4 decimal places). The value returned will override the mask of the current column. Please note that Microdata and Prime systems would use the format (MD14) instead of (MR14).

See the DATE and TIME functions below for examples of using conversions to return the current date and time.

SYSTEM DATE

D - which returns the current system date in the internal format.

Examples: D(D2/) - returns the current system date with the "D2/" conversion applied. This returns the current date in the "mm/dd/yy" format.

SYSTEM TIME

T - which returns the current system time in the internal format.

Examples: T(MTS) - returns the current system time with the "MTS" conversion applied. This returns the current time in the "hh:mm:ss" format.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

RETRIEVING INFORMATION FROM YOUR DATABASE

One of the most powerful features of COMPU-SHEET is its ability to access information in your database and bring it into your spreadsheet for display and/or use in any calculation.

There are several points which must be considered prior to the utilization of this feature. First, the file access flag within COMPU-SHEET must be set to allow reading of your database. Second, your master dictionary must have "Q" pointers set to the files which are to be accessed from COMPU-SHEET. Third, you must know the format of the files you want to read. That is, you must know the file names, item-IDs to be read, and the specific attribute, value, and sub-value numbers you want to retrieve.

The commands that retrieve information from your database are designed to read information from specific items in your files. That means, for example, you can direct COMPU-SHEET to "go to the GL Master file, read account number '12-3456', and return attribute 12". However, for example, you cannot direct COMPU-SHEET to "go to the GL Master file and return the total of all the credit balances in department 42". In the first example, you are specifying a specific item and attribute to retrieve. In the second, you would be asking it to process the file for selected items and conditions. However, this does not mean you cannot accomplish this second example. One method is to use the ACCESS verb "REFORMAT" or "SREFORMAT" if available on your system. Among other things, these verbs provide a means to select items from a file (based upon your selection criteria), accumulate totals, and write these summarized items into another file. This second file can then be accessed by COMPU-SHEET. Please study these verbs, they can be useful in many areas. Another more technical method involves writing small BASIC programs which process a set of items and write a summary total to a file which can then be accessed by COMPU-SHEET. This method is designed to process a large number of items in the same manner your inquiry language can process items.

The following discussions cover the file accessing commands. Note these commands must be surrounded with single quotes.

'READ {DICT} filename exp1 exp2'

This expression will read the file "filename" (or DICT filename) and retrieve the item with an item-ID equal to "exp1" and will return the attribute, value, or sub-value indicated in "exp2".

"exp1" is the item-ID and must be either a location containing the item-ID or the item-ID in literal form (enclosed in double-quotes).

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

If "exp1" is a N (no quotes), the ID will be obtained from the next available item-ID from a select list. This select list must be stored as an item in the spreadsheet file, under the name "spreadsheet name*SELLIST", where the "spreadsheet name" is the name of the current spreadsheet.

"exp2" must be either a location containing the "attribute{,value{,sub-value}}" or the "attribute{,value{,sub-value}}" in literal form (enclosed in double quotes). If only an attribute number is specified, and the attribute contains multi-values, the multi-values will be summed into one value.

'GET exp1'

This expression will retrieve the attribute, value, or sub-value contained in "exp1" from the item read in the previous "READ" statement. It is the equivalent to the "extract" function in BASIC.

"exp1" must be either a location containing the "attribute{,value{,sub-value}}" or a literal containing "attribute{,value{,sub-value}}". If only an attribute number is specified, and the attribute contains multi-values, the multi-values will be summed into one value.

Note: The READ and GET commands will return data from a file. Since COMPU-SHEET stores numeric data with 4 decimal places, it may be necessary to concatenate from 1 to 4 zeroes to the value returned to maintain decimal point alignment.

Examples: 'READ GL "12-3456" "12"':"00" - reads attribute 12 from the item with an ID of "12-3456", in the file named GL. Two zeros are concatenated (appended) to the value read from the file in order to maintain decimal point alignment.

'READ SALES 2.5 3.5':"0000" - reads the file named "SALES". The item-ID to read is to be found as the contents of location 2.5 and the attribute (value, sub-value) is to be found in location 3.5. Four zeros are concatenated to the value read from the file to maintain decimal point alignment.

'GET "13"' - retrieves (extracts) attribute 13 from the last item read by a "READ" command.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

RETRIEVING DATA FROM OTHER SPREADSHEET LOCATIONS

COMPU-SHEET provides the ability to retrieve information from locations in other spreadsheets in the same file as the current spreadsheet. The "/MERGE" command, described earlier in this manual, can be used to copy or merge blocks of locations in other spreadsheets into the current spreadsheet. Another method of retrieving information from other spreadsheet locations is by using the "READW" and "GETW" commands in your formulas. These commands will then be executed each time you calculate your spreadsheet.

The following discussion covers these commands. Please note the commands are surrounded with single quotes.

'READW name loc'

This expression reads a spreadsheet called "name" (in the current file) and returns the location specified in "loc".

"name" may be a location containing the name of the spreadsheet or the name of the spreadsheet in double-quotes.

"loc" may be a location in the current spreadsheet containing the location in the "spreadsheet" or it may be the location in double-quotes.

'GETW loc'

This expression retrieves another location out of the spreadsheet read with the previous "READW" expression.

"loc" may be a location in the current spreadsheet containing the location in the "spreadsheet" or it may be the location in double quotes.

Examples: 'READW "BUDGETS" "2.5"' - reads the spreadsheet named "BUDGETS" in the current spreadsheet file, and returns the contents of location 2.5.

'READW 3.5 3.6' - reads a spreadsheet from the current spreadsheet file. The name of the spreadsheet to read is found in location 3.5 of the current spreadsheet. Location 3.6 of the current spreadsheet contains the number of the location to read from the named spreadsheet.

'GETW "2.6"' - returns the contents of location 2.6 in the spreadsheet read last by a READW command.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

CONCATENATION OPERATORS

The : (colon) operator specifies concatenation of the results of two expressions. It is most commonly used to append zeros to the data read from files in the database.

Examples:

'READ SALES "22-5" "12,5"':"00" - this example reads value 5 of attribute 12 of the item "22-5" in the "SALES" file, and concatenates two zeros to value read. The data stored in the "SALES" file had 2 decimal places, and COMPU-SHEET stores data internally at 4 decimal places. The concatenation of 2 zeros forces decimal point alignment.

RELATIONAL OPERATORS

The operators < > [ ] = and # represent less than, greater than, less than or equal, greater than or equal, equal, and not equal, and denote logical relational operations. These take two expressions as operands and evaluate to either "1" (true) or "0" (false). Relational operators are most often used in the "CONDITIONAL STATEMENT" explained below.

CONDITIONAL STATEMENT

The "IF" statement gives the algebraic format conditional capabilities. The format is as follows:

IF expression THEN expression ELSE expression

The expression associated with the "ELSE" clause may be another "IF" statement. The expression associated with the "THEN" clause will NOT support an "IF" statement. The logical connectives "AND" and "OR" are not supported. Each "IF" statement must have associated "THEN" and "ELSE" expressions. Note this statement is not surrounded with quotes.

Examples:

IF 2.5 < 3.5 THEN 2.5*"1.25" ELSE 3.5 - tests the condition 2.5 less than 3.5. If the condition is true (2.5 is less than 3.5), the formula 2.5*"1.25" will be executed. If the condition is false (2.5 is NOT less than 3.5) the value in location 3.5 will be returned as the result.

IF (2.5-2.6)#2.7 THEN "OUT OF BALANCE" ELSE "" - tests the condition (2.5-2.6) not equal 2.7. If the condition is true, the literal value "OUT OF BALANCE" is returned. If the condition is false, null (nothing) will be returned.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

PRECEDENCE

The precedence of operators is important to keep in mind when writing algebraic formulas. Infinite levels of parentheses are allowed to specify the order of operations. In the absence of parentheses, multiplication and division have greater precedence over addition and subtraction, which in turn, have greater precedence than the relational operators. If two operators have the same precedence, they are applied from left to right. For example, 2.5+3.5+4.5<5.5 will be evaluated as ((2.5+3.5)+4.5)<5.5 and 4.5/5.5*6.5 will evaluate as (4.5/5.5)*6.5.

MATH AND TRIG FUNCTIONS

There are a number of math and trig functions available. These are:

SIN(expression) - returns the sine of the expression in degrees

COS(expression) - returns the cosine of the expression in degrees

TAN(expression) - returns the tangent of the expression in degrees

LN(expression) - returns the natural (base e) logarithm of the expression

EXP(expression) - raises the number "e" (2.1783) to the value of the expression

SQRT(expression) - returns the square root of the expression

PWR(exression,expression) - raises the first expression to the power denoted by the second expression. If the second expression is zero, the value one will be returned

REM(expression,expression) - returns the remainder of the first expression divided by the second expression

ABS(expression) - returns the absolute value of the expression

INT(expression) - returns the integer portion of the specified expression

RD(expression,n) - returns the result of the expression rounded to "n" decimal places

RND(expression) - returns a random number between zero and one less than the expression

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

TABLE LOOKUP FUNCTION

A table lookup function is available which can perform a lookup function within the current spreadsheet or within another spreadsheet. The format is:

'FIND,loc1,loc2,loc3,offset,name'

where:

FIND - is the operation

loc1 - is the location within the current spreadsheet which is to be used as the location to "lookup" (this value can be either absolute or relative)

loc2 - is the location where the lookup is to begin the search (this location must be absolute)

loc3 - is the location where the lookup is to end the search (this location must be absolute)

offset - is the number of columns or rows away from the search list where the value to return is located

name - this value is optional. If not included, the table is to be found within the current spreadsheet. If the "name" is entered, the table is to be found in the current spreadsheet file, in the spreadsheet "name".

HOW THE LOOKUP WORKS

The lookup function will use a table either within the current spreadsheet (no "name" option), or within another spreadsheet (with the "name" option). If the table is to be found in another spreadsheet, the lookup will occur in a spreadsheet with the name "name". The named spreadsheet must be in the current spreadsheet file.

The cell number in "loc1" must be the value in the current spreadsheet which is to be used as the key for the lookup. The cell number in "loc2" and "loc3" define the range of elements in the table. This range can extend down a column or across a row. The value in "offset" indicates where the value to return is located. If the range extends down a column, the offset indicates "how many columns away" is the value to return. If the range to search is in column 5, and the offset is 1, the values in column 6 will be returned.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

The elements of "loc2" thru "loc3" are searched for the largest value that is no greater than the value in "loc1". When a "hit" occurs, the value returned is indicated by the "offset" (the number of columns or rows "away" from the search list). If the value to search for is less than the first entry in the table, "ERR17" is returned.

Examples:

'FIND,2.5,25.10,25.20,1' - this formula searches the current spreadsheet, in column 25 from row 10 thru row 20 for the largest value that is not greater than the contents of location 2.5. If a "hit" occurs, the corresponding row in column 26 (offset 1) will be returned. If the value in 2.5 is less than the value in location 25.10, "ERR17" will be returned.

'FIND,2.5,25.10,25.20,1,"TAX.TABLE"' - this operates exactly as the example above, except that the table to search is in a spreadsheet named "TAX.TABLE" rather than the current spreadsheet.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

FINANCIAL CALCULATION ROUTINE

There is a financial calculation routine available on COMPU-SHEET. This routine will calculate such factors as interest rate, future value, present value, and annuity payments when any one variable is unknown. This routine has been implemented in BASIC and due to the precision limitations on the operating system, must accomplish floating point processes when floating point instructions are not available. This routine can be very slow to execute for some types of calculations. (Timings range from several seconds to up to several minutes, depending upon the type of calculation being performed.) The routine is included for those situations which require the availability of such calculations regardless of the time required.

'FC,exp1,exp2,exp3,exp4,exp5,exp6,exp7'

The "exp" parameters may be either a literal enclosed in double quotes or a location on the spreadsheet containing the parameter. If a parameter is not applicable for a calculation, it should be set to null by entering two commas (,,). Enter a question mark "?" (in double-quotes) in the parameter which is to be solved for.

"exp1" = (S) annuity type - 0=ordinary annuity 1=annuity due

"exp2" = (N) number of periods or years (based upon use of compounding factor

"exp3" = (R) interest rate per period or year (in decimal form..."."12"=12%)"

"exp4" = (F) future value"

"exp5" = (P) present value"

"exp6" = (A) annuity payment (equal payments per period)"

"exp7" = (C) compounding factor (1=annual, 4=quarterly, 12=monthly, 360 or 365=daily)

Note the expression is surrounded with single quotes signs.

ALWAYS DESIGNATE THE DECIMAL PLACE IN EACH AND EVERY ONE OF THE EXPRESSIONS WHERE YOU PUT A NUMERIC VALUE.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

This routine has the ability to solve for number of periods (N), interest rate (R), future value (F), present value (P), or annuity payment (A) according to the following table:

SOLVE FOR:	PARAMETERS: 0 OR 1, REQ=REQUIRED, OPT=OPTIONAL (S)	(N)	(R)	(F)	(P)	(A)	(C)
(N)		?	REQ	REQ	REQ		OPT
(N)		?	REQ	REQ		REQ	OPT
(N)		?	REQ		REQ	REQ	OPT
(R)		REQ	?	REQ		REQ	OPT
(R)		REQ	?	REQ	REQ		OPT
(R)		REQ	?		REQ	REQ	OPT
(F)		REQ	REQ	?		REQ	OPT
(F)	0	REQ	REQ	?	REQ		OPT
(F)	1	REQ	REQ	?	REQ		OPT
(P)		REQ	REQ	REQ	?		OPT
(P)	0	REQ	REQ		?	REQ	OPT
(P)	1	REQ	REQ		?	REQ	OPT
(A)	0	REQ	REQ	REQ		?	OPT
(A)	1	REQ	REQ	REQ		?	OPT
(A)	0	REQ	REQ		REQ	?	OPT
(A)	1	REQ	REQ		REQ	?	OPT

Examples:

´FC,,"30.","?",,2.,3.,"12."´ - solves for the annual interest rate of a 30 year loan. The original loan amount is in column 2 (on the row containing the formula), and there are 12 payments per year in the amount in column 3 (on the row containing the formula).

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

CALLING SUBROUTINES FROM COMPU-SHEET

A BASIC subroutine can be called directly from any formula. This feature offers those users with programming skills the ability to define special routines which can be called from COMPU-SHEET to perform special tasks.

Subroutines can be used to perform special file processing, specialized calculation routines, interface to existing application software, or any number of functions unique to a business.

CALLING A SUBROUTINE FROM A FORMULA

The following format is used to call a subroutine:

```
'@name'
```

where:

'	- the call must be surrounded with single quote signs
@	- indicates a subroutine call
name	- is the name of the cataloged subroutine

Values can be passed to the subroutine with the following format:

```
exp,exp,exp...,'@name'
```

where:

exp	- can be any valid spreadsheet location or literal
'@name'	- is the subroutine call as described above

CODING THE BASIC SUBROUTINE

The BASIC subroutine must have the following statement on line 1:

```
001 SUBROUTINE name(COL,ROW,STACK,DATA,WORK,CALCS,COLROW,FNAME,SNAME)
```

The variables are:

name	- the subroutine name
COL	- the column number containing the formula
ROW	- the row number containing the formula
STACK	- the "STACK", a dynamic array containing data passed to and from the subroutine
DATA	- the current DATA dynamic array
WORK	- the current WORK dynamic array
CALCS	- the current CALCS dynamic array
COLROW	- the current COLROW dynamic array
FNAME	- the name of the spreadsheet file (DO NOT CHANGE)
SNAME	- the name of the spreadsheet

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

Examples:

The formula calling the subroutine is: 2.5,"3.33",2.6,'@CS.SUB'

The BASIC code for the subroutine is:

```
001 SUBROUTINE CS.SUB(COL,ROW,STACK,DATA,WORK,CALCS,COLROW,FNAME,SNAME)
002 A=STACK<1> ; * pulls the value in location 2.6 from the STACK
003 B=STACK<2> ; * pulls the literal value "3.33" from the STACK
004 C=STACK<3> ; * pulls the value in location 2.5 from the STACK
005 D=(a formula that computes the value of D using A B and C)
006 STACK=D    ; * sets stack equal to the result (D)
007 RETURN     ; * returns control back to COMPU-SHEET
```

The contents of STACK will display as the result of the formula.

Any data passed to the subroutine will be placed as attributes in "STACK". The result of your subroutine should be placed into attribute 1 of STACK. REMEMBER: COMPU-SHEET performs all calculations with four decimal place precision. Numeric data in the STACK and DATA arrays will have four implied decimal places.

Refer to the section on "TECHNICAL INFORMATION" for further information on the dynamic arrays passed to the subroutine.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - REFERENCING OTHER FORMULAS - -

Because of the method in which formulas are handled by COMPU-SHEET, it is common for the same formula to be found in a large number of locations within a spreadsheet. That is, a formula can be entered into one location and then subsequently repeated down a column or across a row. This method of entering formulas is effective; however, an alternative method is available which is more efficient. This method involves referencing a formula contained in one location from another location. To explain this concept further, imagine a spreadsheet which contains a series of columns, each of which is totalled at the bottom of the spreadsheet. This formula could have been written:

```
'A,.5,.15'
```

which accumulates rows 5 thru 15 within a column. Let's assume this formula is at location 2.17 and columns 3 thru 13 are to have totals on row 17 as well. Locations 3.17 thru 13.17 could each contain the formula:

```
$2.17
```

which means "use the formula in location 2.17". This method saves a considerable amount of space in storing formulas and makes it much easier to change formulas within a spreadsheet.

The format is as follows:

$col.row	Where "col.row" is the standard format defined under the topic "SPREADSHEET LOCATIONS"

This formula can be entered in any cell at the "FORMULA:" prompt as any other formula. The "/REP" command has an option to automatically create the "$" reference at the time the "/REP" command is executed.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - ALGEBRAIC FORMULA EDITOR - -

COMPU-SHEET has an editing capability for the algebraic formulas to make display and modification of formulas somewhat easier. The editor commands are entered at the "FORMULA:" prompt. These commands are:

FORMULA: DF

The "DF" command will cause the formula to be displayed on the status lines. This gives the user the ability to display formulas too large to fit within the normal formula display area.

FORMULA: R/aaa/bbb/n

The "R" (Replace) command allows the modification of a formula without completely reentering it. The format is as follows:

"R" indicates replace

"/" is any delimiter (any special character: /,;.? etc.)

"aaa" are the characters to be replaced with "bbb"

"bbb" are the characters to replace "aaa"

"n" is the occurance of "aaa" to be replaced by "bbb" or an "*" which indicates all occurances of "aaa" are to be replaced with "bbb". For example: if "n"=2 then only the second occurance of "aaa" will be replaced with "bbb". If the final "/n" is not entered, the first occurance is assumed.

As each execution of the Replace command is completed, the modified formula will be displayed on the status lines. When all modifications have been completed, <CR> at the "FORMULA:" prompt will cause the formula to be recalculated and the results to be displayed.

FORMULA: Cloc

The "C" (Copy) command, provides a quick means to copy a formula from another location into the current location. The value "loc" can be any valid spreadsheet location.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - EXAMPLES OF ALGEBRAIC FORMULAS - -

FORMULA: ((3.5+3.6)*"1000.")/2.7
This formula adds location 3.5 to location 3.6 and the result is multiplied by the literal value "1000.". The result of that is then divided by location 2.7.

FORMULA: D(D2/)
This formula takes todays date ("D") and converts it to "mm/dd/yy" ("D2/") format.

FORMULA: IF 4.5+5.5 < "100." THEN "UNDER LIMIT" ELSE "OVER LIMIT"
This formula takes the sum of locations 4.5 and 5.5 and tests the result to see if it is less than "100.". If it is, the value of the formula is "UNDER LIMIT". If not, the value of the expression is "OVER LIMIT"

FORMULA: IF 10.3=5.2 THEN "" ELSE IF 10.3=5.3 THEN 6.2 ELSE "CHECK"
This is a multiple "IF" statement. It checks to see if location 10.3 is equal to location 5.2. If this test is equal, then the value of the formula is null. If the test was not equal, then a second test is made, to see if location 10.3 is equal to location 5.3. If the second test is equal, then the value of the formula is the same as location 6.2. If not, the value of the formula is "CHECK".

FORMULA: 3.5+'A,2.3,2.10'
This formula adds location 3.5 to the accumulated value of row 3 thru row 10 in column 2.

FORMULA: 3.5["1","20"]
This formula takes the contents of location 3.5 and extracts the first 20 characters.

FORMULA: "**********"["1",3.4]
This formula retrieves the string of "*" starting with the first "*" for a count equal to the value in location 3.4. For example, if location 3.4 had a value of "12", then the value of this formula would be "**********" (12 asterisks). This type of formula is used to create bar graphs with COMPU-SHEET.

FORMULA: 'READ GL "12-345" "8,2"':"0000"
This formula reads the item "12-345" from the file "GL" and retrieves attribute 8, value 2. Four zeroes are concatenated to the returned value to adjust it to the internal value used by COMPU-SHEET.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

FORMULA: ´GET "9"´:"00"
This formula retrieves attribute 9 from the item read by the last executed "READ" statement. If attribute 9 consisted of multiple values, they would be summed into a single value. Two zeroes are concatenated to the value returned.

FORMULA: ´READW "BUDGET1" "9.15"´+5.12
This formula reads the spreadsheet called "BUDGET1" and retrieves location 9.15. The retrieved value is then added to location 5.12 on the current spreadsheet.

FORMULA: (´GETW "9.16"´+´GETW "9.17"´)*2.37
This formula adds locations 9.16 and 9.17 from the last spreadsheet read by a "READW" command and then multiplies the result times the value of the current spreadsheet location 2.37.

FORMULA: ´FC,,"3.",".16",,"20000.","?","12."´
This formula will calculate the annuity payment amount based upon a present value of $20,000.00, over 3 years, at 16% interest, with a compounding factor of 12.

FORMULA: ´A,.5,.P1´
This formula accumulates rows 5 thru one row prior to the formula within the current column (the column containing the formula).

FORMULA: ´AVG,5.4,10.15´
This formula will return the average of all numeric values for all locations within the range of column 5 thru column 10, rows 4 thru 15.

FORMULA: .3,.4,´@CHKSUB´
This formula places the contents of row 3 and row 4 (in the current column) into the stack, and calls the subroutine "CHKSUB".

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - FUNCTION PROCESSOR FORMAT - -

The function processor format is an ALTERNATIVE to the algebraic format. The algebraic format is sufficient for most spreadsheet applications; however, the function processor format is provided for those who wish to use its method.

The function processor format is used to compute a value by performing indicated mathematic and logic operations on one or more operands. The operands may be constants, locations, or special functions. Operand values are stored in an unlimited-entry pushdown stack designated STACK1 (top of stack), STACK2, STACK3, ...STACKn.

The general format of the function is:

step\{step\step}....

The function must contain at least one backslash (\) character following the first step. Each "step" must be separated by a backslash character (\). A "step" may be any one of the following:

1. A location reference specifying a value in a spreadsheet location to be pushed onto the stack.

2. A constant in the form of Cn where "n" is a numeric or string constant to be pushed ontc the stack.

3. Any valid conversion (input or output).

4. A special function which may accumulate a range of values, read an item from a file in the system, write an item to a file in the system, or some other special type of function.

5. An operator which specifies an operation to be performed on the top entries on the stack.

6. An operator which instructs the function processor to "jump" or skip over a specified number of steps.

Steps may cause values to be pushed onto the stack, with existing values (if any) moved down one position in the stack. Likewise, some steps may cause one or more values in the stack to be "popped" off the stack, with any remaining values moved up accordingly.

The entry remaining in STACK1 when the last step has been executed will be the value for the location being calculated.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

THE STACK

A pushdown stack is used to perform function processor operations.

Operations specified by the function processor operate on the top entries in a pushdown stack. This pushdown stack has a 32k character capacity and can be visualized as follows:

```
STACK1 -> [______]
STACK2 -> [______]
STACK3 -> [______]
STACK4 -> [______]
STACK5 -> [______]
STACK6 -> [______]
STACK7 -> [______]
STACK8 -> [______]
    and so on....
```

STACK1 is the top position in the stack, STACK2 is the next position, etc. As a value is pushed into the stack, it is pushed into position STACK1; the original value of STACK1 is pushed into STACK2 and so on. As a value is fetched off the stack, it is popped from position STACK1, the original value of STACK2 moves up to STACK1, and so on.

The function comprises any number of operands or operators (steps) in reverse Polish format, separated by backslashes (\). When the function processor encounters an operand specification, it "pushes" the corresponding value onto the top of the stack (STACK1). When the function processor encounters an arithmetic operator, it performs the corresponding operation on the top entries in the stack. When the entire function has been computed, the top entry in the stack (STACK1) will be the value retrieved and placed into the location being calculated.

For a more complete description of a stack processor, refer to your system manuals, under the topic "F CODE STACK".

WORKAREA

There is a special "workarea" available to the user. This workarea is used as the internal storage area for any items which are read from other files, or will be written to other files. This workarea is accessed by several of the "special functions" available in the function processor.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

STEPS

The following explains each valid operation available in the function processor. Note all arithmetic calculations maintain 4 decimal places.

STEP	DESCRIPTION
c1.r1	Where "c1" is a column number and "r1" is row number
c1.	Where "c1" is a column number and row is current row
.r1	Where "r1" is a row number and column is current column
	Note: In all cases "c1" and "r1" may be a relative location in the Pn or Fn format. See the section "SPREADSHEET LOCATIONS" for more information.
	Push corresponding location value onto pushdown stack: Location value -> STACK1 -> STACK2 -> STACK3 ->...etc.
(conversion)	Apply output conversion to STACK1
	STACK1(output conversion) -> STACK1 (no change in other stack entries)
(I,conversion)	Apply input conversion to STACK1
	STACK1(input conversion) -> STACK1 (no change in other stack entries)
Cn	Constant
	Push numeric or string constant "n" onto stack: "n" -> STACK1 -> STACK2 -> STACK3 ->...etc.
D	Date
	Push current system date in internal format onto stack: date -> STACK1 -> STACK2 -> STACK3 ->...etc.
T	Time
	Push current system time in internal format onto stack: time -> STACK1 -> STACK2 -> STACK3 ->...etc.
+	Addition
	STACK1 + STACK2 -> STACK1 ; STACK2 <- STACK3 <-...etc.
-	Subtraction
	STACK1 - STACK2 -> STACK1 ; STACK2 <- STACK3 <-...etc.
*	Multiplication
	STACK1*STACK2 -> STACK1 ; STACK2 <- STACK3 <-...etc.
/	Division
	STACK1 / STACK2 -> STACK1 ; STACK2 <- STACK3 <-...etc.
R	Remainder
	REM(STACK1/STACK2) -> STACK1 ; STACK2 <- STACK3...

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

S	Summation of multi-values summation(STACK1) -> STACK1 Prior to this operation, STACK1 may be multi-valued (from reading an attribute from another file); this operator sums all the multi-values into a single value
E	Raises STACK1 to the power of STACK2 PWR(STACK1,STACK2) -> STACK1; STACK2 <- STACK3 <-...
s	Returns the sine of STACK1 in degrees SIN(STACK1) -> STACK1
c	Returns the cosine of STACK1 in degrees COS(STACK1) -> STACK1
t	Returns the tangent of STACK1 in degrees TAN(STACK1) -> STACK1
l	Returns the natural (base e) logarithm of STACK1 LN(STACK1) -> STACK1
e	Returns the number 'e' (2.1783) to the value of STACK1 EXP(STACK1) -> STACK1
q	Returns the square root of STACK1 SQRT(STACK1) -> STACK1
a	Returns the absclute value of STACK1 ABS(STACK1) -> STACK1
i	Returns the integer value of STACK1 INT(STACK1) -> STACK1
r	Returns a random number between zero and one less than STACK1 RND(STACK1) -> STACK1
d	Returns STACK2, rounded to the number of decimal places indicated in STACK1 ROUND(STACK2,STACK1) -> STACK1; STACK2 <- STACK3... ABS(STACK1) -> STACK1
"	Duplication STACK1 -> STACK2 -> STACK3 -> STACK4 ->...etc.
_	Exchange STACK1 <-> STACK2
^	"Pop" stack STACK1 <- STACK2 <- STACK3 <- STACK4 <-...etc.
:	Concatenation STACK1:STACK2 -> STACK1 ; STACK2 <- STACK3 <-...etc.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

[] Text extraction
STACK3[STACK2,STACK1] -> STACK1 ; STACK2 <- STACK4 ;
STACK3 <- STACK5 ; STACK4 <- STACK6 ; ...etc.

= Equal
1) If STACK1 = STACK2 then 1 -> STACK1 (See note)
2) If STACK1 # STACK2 then 0 -> STACK1 (See note)

Not equal
1) If STACK1 # STACK2 then 1 -> STACK1 (See note)
2) If STACK1 = STACK2 then 0 -> STACK1 (See note)

< Less than
1) If STACK1 < STACK2 then 1 -> STACK1 (See note)
2) If STACK1 => STACK2 then 0 -> STACK1 (See note)

> Greater than
1) If STACK1 > STACK2 then 1 -> STACK1 (See note)
2) If STACK1 <= STACK2 then 0 -> STACK1 (See note)

[Less than or equal
1) If STACK1 <= STACK2 then 1 -> STACK1 (See note)
2) If STACK1 > STACK2 then 0 -> STACK1 (See note)

] Greater than or equal
1) If STACK1 => STACK2 then 1 -> STACK1 (See note)
2) If STACK1 < STACK2 then 0 -> STACK1 (See note)

Note: in each case STACK2 <- STACK3 <- STACK4 <-...etc.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

SPECIAL FUNCTIONS

STEP	DESCRIPTION
;expression	Execute algebraic expression and push on STACK1 STACK1 -> STACK2 -> STACK3 ->...etc.
?	If STACK1 = 1 then execute next step else skip next step in function STACK1 <- STACK2 <- STACK3 <-...etc.
Jn	Jump or "skip" over next "n" steps in function (no change in stack)
L,loc	Push contents of STACK1 onto location "c1.r1" (Note: the spreadsheet must be redisplayed in order to cause the location to display) STACK1 <- STACK2 <- STACK3 <-...etc.
A,loc1,loc2	Accumulate locations "loc1" thru "loc2" and push on STACK1 STACK1 -> STACK2 -> STACK3 ->...etc.
&AVG,loc1,loc2{,S}	Average locations "loc1" thru "loc2" and push on STACK1 (S indicates omit zero values). STACK1 -> STACK2 -> STACK3 ->...etc.
&CNT,loc1,loc2{,S}	Count locations "loc1" thru "loc2" and push on STACK1 (S indicates omit zero values). STACK1 -> STACK2 -> STACK3 ->...etc.
&MIN,loc1,loc2{,S}	Return minimum value of "loc1" thru "loc2" and push on STACK1 (S indicates omit zero values). STACK1 -> STACK2 -> STACK3 ->...etc.
&MAX,loc1,loc2{,S}	Return maximum value of "loc1" thru "loc2" and push on STACK1 (S indicates omit zero values). STACK1 -> STACK2 -> STACK3 ->...etc.
N	READNEXT ID from SELECT list and push on STACK1 (at end place "END" on STACK1); A SELECT, SSELECT, GET-LIST, or FORM-LIST must be executed prior to executing COMPU-SHEET; STACK1 -> STACK2 -> STACK3 ->...etc.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

R,{DICT,}filename	Read {DICT} filename into STACK1 ; prior to execution: STACK1 = ID STACK2 = amc after execution: data -> STACK1 ; STACK2 <- STACK3 <-...etc.
W,{DICT,}filename	Write the contents of STACK3 onto {DICT} filename using STACK1 as the ID and STACK2 = amc after execution: STACK1 <- STACK4 ; STACK2 <- STACK 5 <-...etc.
!READ,{DICT,}filename	Read {DICT} filename into WORKAREA ; prior to execution: STACK1 = ID after execution: STACK1 <- STACK2 <- STACK3 <-...etc.
!WRITE,{DICT,}filename	Write {DICT} filename from WORKAREA ; prior to execution: STACK1 = ID after execution: STACK1 <- STACK2 <- STACK3 <-...etc.
G	Retrieve amc or value or sub-value from WORKAREA ; prior to execution: STACK1 = amc{,value-mark{,sub-value mark}} after execution: STACK1 = data (no change in stack)
H	Place STACK2 into amc, value, or sub-value of WORKAREA ; prior to execution: STACK1 = amc{,value-mark{,sub-value mark}} STACK2 = data to place in WORKAREA after execution: STACK1 <- STACK3 <- STACK4 <-...etc.
@subroutine	Causes subroutine "subroutine" to be called. Elements passed to the subroutine are: COL,ROW,STACK,DATA,WORK,CALCS,COLROW Line 1 of the "subroutine" should be: "SUBROUTINE name(COL,ROW,STACK,DATA,WORK,CALCS, COLROW,NAME,ANAME)" (See the section on "TECHNICAL INFORMATION")

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

EXAMPLES:

FORMULA: 2.5\2.6\2.7\+\+

FORMULA: D\(D2/)

FORMULA: 3.5\(I,D)\4.5\(I,D)\+(D)

FORMULA: 3.4\!READ,CM.FILE\C1,2,3\G

FORMULA: 2.10\3.10_\?\J2\CGREATER\X\CLESS

FORMULA: A,3.,8.\2.5*

FORMULA: 3.\4.\5.\+\-

FORMULA: ;.2+.3+.4\@SUBA010\.5*

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

FUNCTION EDITOR

The input routine for the function processor prompts with: "FORMULA:". COMPU-SHEET has a limited editing capability to make the entry of functions somewhat easier. This editor will allow insertion, replacement, deletion, copying, and display of function steps. The editor commands are entered at the "FORMULA:" prompt. These commands are:

FORMULA: In — where "I" represents "insert" and "n" is the step number which follows the step to be inserted. Upon entry, the Command: line will request: "Insert::". At this point, enter one step to be inserted into the formula.

FORMULA: Rn — where "R" represents "replace" and "n" is the step number to replace. Upon entry, the Command: line will display the step to be replaced and will prompt for the new step with "To::". At this point, enter one step to replace the old step.

FORMULA: Dn — where "D" represents "delete" and "n" is the step number to delete. Upon entry the step will be deleted.

FORMULA: Cc1.r1 — where "C" represents "copy" and "c1.r1" is the location where the formula is to be copied from. Upon entry, the copy will be executed.

FORMULA: DF — the "DF" command will cause the formula to be displayed on the status lines. This gives the user the ability to display formulas too large to fit within the normal display.

In all cases, the command will be executed and the command line will return to the "FORMULA:" prompt for any additional modifications.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - TECHNICAL INFORMATION - -

The information contained in this section is NOT necessary for most users of COMPU-SHEET, unless they intend to become involved in more technical aspects - such as writing subroutines or a program interface with existing application software.

COMPU-SHEET has been written entirely in PICK/BASIC except for the screen display routine, which has been written in assembly language for speed considerations (however, the BASIC equivalent for this routine is also included). You will find COMPU-SHEET to be quite structured in design. It consists of one main controlling program and nine subroutines which are called as needed depending upon the command currently being executed. This documentation will overview the file structure, and the programs and subroutines.

THE COMPU-SHEET FILE STRUCTURE

COMPU-SHEET requires up to four files during the execution of the spreadsheet. These files are:

spreadsheet file - this file is the file selected by the user to store the spreadsheet.

CS.CONTROL - this file contains the terminal control parameters and COMPU-SHEET prompts.

CS.HELP - this file contains the COMPU-SHEET help messages.

filbld file - this is any user selected file which may be used to receive a COMPU-SHEET spreadsheet from the /FILBLD process.

The following sections will review each of these files.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

THE SPREADSHEET FILE

The spreadsheet file contains up to five items for each spreadsheet created by COMPU-SHEET. Each of these five items has an item-ID which consists of the spreadsheet name followed by an "*" followed by the name of the item, "COLROW", "DATA", "CALCS", "WORK", or "SELLIST". The format of these items is as follows:

"item.name*COLROW"

ATTRIBUTE	CONTENTS
1	A multi-valued list of column lengths
2	A multi-valued list of column justifications (L or R)
3	A multi-valued list of column masks (MR2, or MR0,E...)
4	A multi-valued list of column separators
5	A multi-valued list of window starting and ending columns where sub-value 1 is the starting column and sub-value 2 is the ending column
6	A multi-valued list of window starting rows
7	The last selected window
8	A multi-valued list of window row number suppress codes where null=print and 1=suppress
9	Is the spreadsheet password
10	A multi-valued list (up to 10) of spreadsheet description
11	A multi-valued list where: value 1 = operator intitials value 2 = date of last update value 3 = time of last update
12	A multi-valued list where: value 1 = prompt direction value 2 = calculation direction value 3 = page toggle
13	Is the number of the last row in the heading lock
14	Is the cell protection and next location definition array where each value corresponds to a row and each sub-value corresponds to a column and a sub-value of "1" indicates the cell is protected and a null sub-value is an unprotected cell and a sub-value containing a location indicates the next location for data entry
15	A multi-valued list of macro names
16	A multi-valued list of macros (corresponding to attr 15) where each sub-value is a step in the macro

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

"item.name*DATA"

Each attribute corresponds to a row and each value corresponds to a column. The contents of a value is the unmasked data displayed or printed on the spreadsheet. All numeric data have four implied decimal positions.

"item.name*CALCS"

Each attribute corresponds to a row and each value corresponds to a column. The contents of a value is the formula associated with this cell. The formula consists of sub-value 1 which is the formula in algebraic format (if any) and sub-values 2 to n are the parsed formula in stack processor format.

"item.name*WORK"

This array may contain one of several items. If another spreadsheet is read from this spreadsheet, the work array contains the data array from the spreadsheet that was read. If this spreadsheet reads a data item from any other file, the item is read into this work array. Also, if the user desires, the work array may be used as a work area while using the function formula format.

"item.name*SELLIST"

This array is supplied by the user and contains any select lists which are to be processed by the spreadsheet.

The list can be a list created by a SELECT or SSELECT and copied into the file under the item-ID "item.name*SELLIST", or it could be created by an application program. The format must be one item-ID per attribute.

This list is accessed during the execution of the "READ" command when the item-ID variable is an "N" (without the quotes) as in the the formula: ´READ GL-MAST N "12"´ which reads the GL-MAST file, using the item-ID´s from the SELLIST item.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

THE CS.CONTROL FILE

The CS.CONTROL file contains several types of items used by COMPU-SHEET.

1) The "TERM.CTL" item contains the table displayed by the CS.TERM program.

2) The "TERM.STD" item contains a set of standard parameters to use as the default when no terminal definition exists for either the port or the terminal type.

3) There are one or more items with item-ID's matching the port number and/or the terminal type. This item defines the visual attributes and printer characteristics for the associated terminal. Also, this item contains user-defined equivalents to COMPU-SHEET commands.

4) There is a complete set of prompts used by COMPU-SHEET. Each prompt displayed by the program is file driven and can be customized. The format of the item is:

```
item-ID = "P*" concatenated to the prompt number
attr 1  = a multi-valued list of prompts where sub-values 1 and 2 are
          the cursor address and the third sub-value is the prompt
attr 2  = a multi-valued list of responses where each sub-value is
          an equivalent response (optional)
attr 3  = a multi-valued list of values to return which corresponds to
          the values in attr 2 (optional)
```

If the prompt number has a value (such as F or D) concatenated to the end, it is an indication that the prompt is for a language other than English.

5) The item "P*SCR" contains the screen literal definitions. Again, if another character is concatenated to the end, it will be considered the language code.

6) The item "P*AUDIT" contains the heading information for the printed audit report. The language code is applicable here too.

7) The item "P*CMND" is optional and may contain a list of synonym commands. Attribute 1 would be a multi-valued list of synonym commands and attribute 2 would contain a list of corresponding valid COMPU-SHEET commands. Again, there may be several of these items, each with a different language code concatenated to the end.

In all cases above where language codes are used, if a language code is used that cannot be found concatenated to one of the above items, the standard item (with no language code) will be retrieved.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

THE CS.HELP FILE

The CS.HELP file contains the online "help" documentation. The item-ID of the items in this file are related to the prompt number from which they are called. For example, help ID "275" corresponds to prompt "P*275" in the CS.CONTROL file. However, help messages may have other associated help messages which make up the extended help function. For example, help number "275" may have help number "275*T" and other messages associated with it. The format of these items are:

item-ID = corresponds to the prompt number

attr 1 = a multi-valued list of help messages where each value corresponds to a line of the help message

attr 2 = a multi-valued list of responses where each value corresponds to a "goto" help message in attr 3 (optional)

attr 3 = a multi-valued list of "goto" message numbers corresponding to the values in attribute 2 (optional)

As above, a language code may be concatenated to the help items.

THE FILBLD FILE

COMPU-SHEET gives the user the option to "dump" any spreadsheet to a standard PICK file. This process will create the dictionary (if the user specifies) as well as the data levels. The dictionary must contain an item called "CS%FILBLD" with an "X" in attr 1. Attributes 2 thru n may contain the names of the spreadsheets which will be allowed to generate items in the file. If attribute 2 is not utilized, any spreadsheet will be allowed to generate items in the file (as long as the "CS%FILBLD" item exists).

One item will be created for each row specified to be output. The item-ID is the row number and there is one attribute for each column specified to be output. If requested, the dictionary will be built, and the dictionary items will be created from the information available for each column.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

THE COMPU-SHEET PROGRAMS

COMPU-SHEET consists of one main control program and nine subroutines which are called depending upon the type of operation requested by the user.

These programs are structured in their approach and should be quite easy to follow by any experienced PICK programmer. Each program contains comment lines describing the function of each main segment of code. The following is a brief discussion of the main program and the functions of each of the subroutines.

PROGRAM CS.MAIN

This is the main controlling program. The functions performed here are:

Accepting the user's command and directing control to the appropriate subroutine

All cursor movement commands

The /DIR, /CDIR, /PAGE, /DIS, =macro, and the data and formula entry commands

The complete display of the spreadsheet and the status lines

SUBROUTINE CS.S1

This subroutine performs the calculations on the spreadsheet. All calculations except the financial calculation routine are handled by this subroutine.

The main segments of the subroutine are the stack processor which handles the actual execution of the formulas, and the "A to F" conversion routine which parses the algebraic formula entered by the user into a stack format for execution by the stack processor.

SUBROUTINE CS.S2

This subroutine handles the spreadsheet initialization. This consists of prompting for the file name, spreadsheet name, password, description, initials, and the handling of the initial creation of the spreadsheet. Also, this routine controls the display or printing of the list of spreadsheets contained in the selected file.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

SUBROUTINE CS.S3

This subroutine handles eight commands. They are:

/WINDOW - the quick window set routine
/SET - the expanded window set routine
/SEL - the command to select a window
/SUPP - the command for suppressing window row numbers
/SETH - the routine that locks heading rows on the screen
/SIZE - the routine that returns the size of the spreadsheet items
/PROT - the routine that provides for cell protection against data entry
/NEXT - the command that directs the data entry process to the next cell

SUBROUTINE CS.S4

This subroutine handles four routines. They are:

/WIDTH - the routine that changes column widths
/JUST - provides for setting/changing column justification and masking
/INS - provides for inserting columns and rows into a spreadsheet
/DEL - provides for deleting columns or rows from a spreadsheet

SUBROUTINE CS.S5

This subroutine processes seven COMPU-SHEET commands. They are:

/COPY - the command for copying columns or rows to other columns or rows
/REP - provides for repeating a location across a row or down a column
/CLR - the routine used for clearing a range of cells
/MERGE - provides for merging other spreadsheets into the current one
/SORTB - the routine for sorting a block of cells on a spreadsheet
/SORT - the routine for sorting a set of rows on a spreadsheet
/MACRO - provides the ability to record and execute a series of commands

SUBROUTINE CS.S6

This subroutine handles the filing and printing of a spreadsheet.
There are six commands processed by this routine. They are:

/FI & /FS- provides for writing the spreadsheet into the file
/TERM - this command defines terminal and printer sizes
/PRINT - prints the spreadsheet on the line printer
/PRINTT - prints the spreadsheet on a terminal slave printer
/AUDIT - prints an audit report that documents the spreadsheet
/AUDITT - prints the audit report on a terminal slave printer

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

SUBROUTINE CS.S7

This subroutine handles the displaying of all help messages.

SUBROUTINE CS.S8

This subroutine is the financial calculation routine. It has been written in BASIC and overcomes the precision problems by performing specialized floating point routines during the calculations.

SUBROUTINE CS.S9

This routine is the /FILBLD command which handles the "dumping" of a spreadsheet into a standard PICK file. This routine will build the dictionary (if selected) as well as the data file.

ASSEMBLY LANGUAGE DISPLAY ROUTINE

A display routine written in assembly code is provided which significantly speeds up the screen display process. The use of this routine is optional. You may choose to use it or, if you prefer, you can use the BASIC version which is also provided. The PROC that calls COMPU-SHEET contains a parameter that indicates the use of the assembly routine or the BASIC routine.

If you choose to use the assembly routine, when COMPU-SHEET is loaded on your computer you must indicate the version of this routine to load for your machine. Both the source and assembled object code are provided. You have the option to specify the system mode where the display routine is to be loaded.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

THE COMPU-SHEET CALLING PROC

The COMPU-SHEET PROC calls CS.MAIN and passes it several parameters. The format is:

```
001 PQ
002 HCS.MAIN
003 STON
004 Hn< - where "n" contains the file accessing flag
005 Hn< - where "n" contains the assembly code control flag
006 Hn< - where "n" indicates command passing from PROC
007 P
```

This PROC contains three parameters which control the operation of COMPU-SHEET. These parameters are:

line 004 - contains the file accessing flag
- 0 = the user is not permitted to read or write to the database
- 1 = the user is permitted to only read the database
- 2 = the user is permitted to read or write to the database

Note: The control over reading and writing to the database applies only to formulas. The ability to write back to the database exists only with the function format.

Line 005 - contains the assembly code control flag
- 0 = use the BASIC display routine only
- 1 = use the assembly language display routine

line 006 - indicates the PROC will be passing commands to COMPU-SHEET
- 0 = the PROC will be passing commands to COMPU-SHEET
- 1 = the PROC will not be passing commands to COMPU-SHEET

Note: If COMPU-SHEET commands will be passed thru the PROC, this value must be set to "0". In normal operation, this value would be set to "1".

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

- - DEFINING TERMINAL CHARACTERISTICS: CS-TERM - -

COMPU-SHEET can be used with all terminals that are available for use on the PICK Operating System. A special routine is available for defining the characteristics of terminals to be used with COMPU-SHEET. These characteristics include utilization of visual attributes such as reverse video, half-intensity, etc.. Also, function keys (or any key on the keyboard) can be equated to a COMPU-SHEET command. This routine provides the user with the ability to take full advantage of any special features available on terminals used with COMPU-SHEET.

The file "CS.CONTROL" contains a set of items which define these characteristics by port and/or terminal type. When COMPU-SHEET is called, it attempts to locate an item in the CS.CONTROL file with an ID equal to the port number of the terminal. If found, the parameters in the item are used to control the operation of COMPU-SHEET. If this item is not found, an attempt will be made to locate an item matching the system TERM type for this port. If still not found, an item called "TERM.STD" is read from the file. This item contains the default parameters.

The routine is called by entering "CS-TERM" from TCL. The process responds with the message:

COMPU-SHEET terminal definition process for port # or term id:

This prompt is requesting either a terminal type code (A, D, L, M, R, etc...) or a port number (0, 1, 2, 3, etc...). These codes can be user-defined and are used to establish parameters for a particular terminal type. Normally, a terminal type code will be used to define the terminal characteristics. However, if preferred, the user can define a unique set of parameters for any port.

After responding to the above prompt, the process will try to locate an item in the CS.CONTROL file with an ID equal to the value entered. If one is found, the current parameter settings are displayed. If it is a new item, the following prompt will appear:

This is a new definition, enter name of existing format to copy,
If desired...:

Here, you may enter the name of any previously defined process and it will be copied to this definition. This will save time in setting up a new definition when one has already been created for a terminal type or port number.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

At this point, the current parameters are displayed. If this is a new definition, each parameter will be requested. Is this is an existing definition, any parameter may be changed as needed. The screen appears as follows:

```
COMPU-SHEET terminal definition process for port # or term id:
    1 Visual attribute flag....(0 or 1 or 2).....
    2 Visual attribute set.....(in hex format)...
    3 Border ON................(in hex format)...
    4 Border OFF...............(in hex format)...
    5 Status line 1 ON.........(in hex format)...
    6 Status line 1 OFF........(in hex format)...
    7 Status line 2 ON.........(in hex format)...
    8 Status line 2 OFF........(in hex format)...
    9 Command line ON..........(in hex format)...
   10 Command line OFF.........(in hex format)...
   11 Window cursor ON.........(in hex format)...
   12 Window cursor OFF........(in hex format)...
   13 AUX port ON..............(in hex format)...
   14 AUX port OFF.............(in hex format)...
   15 Printer characters/line..(in decimal form).
   16 Printer lines/page.......(in decimal form).
   17 Screen width characters..(in decimal form).
   18 Go to command definition screen
```

The following is an explanation of each parameter:

1 Visual attribute flag....(0 or 1 or 2).....

The value indicates this reverse video format for the terminal being defined. A value of "0" indicates this terminal does not support reverse video. A value of "1" indicates this terminal does support reverse video and the reverse video attributes are nonimbedded (that is, the control sequences which initiate and terminate the reverse video do not physically occupy a position on the screen). A value of "2" indicates this terminal does support reverse video and the reverse video attributes are imbedded (that is, the control sequences which initiate and terminate the reverse video physically occupy a position on the screen).

2 Visual attribute set.....(in hex format)...

Some terminals (such as the Ampex D-80 and the ADDS Viewpoint A-2) require a visual attribute initiating sequence. This parameter reflects this sequence. For example: the Viewpoint requires an "ESC:0:P" for initiating full-intensity reverse video, and the D-80 requires "ESC:A". Since the parameter must be entered in hex format, this parameter would be "1B3050" for the Viewpoint and "1B41" for the D-80. Note: "1B" is hex for "ESC", "30" is hex for "0", "50" is hex for "P", and "41" is hex for "A".

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

Parameters 3 through 12 control the visual attribute definitions of various sections of the COMPU-SHEET screen. The following explains the areas controlled by these parameters.

BORDER - is the border defining the column and row numbers, "help" windows, and the set-up screens
STATUS LINE 1 - is the first status line below the spreadsheet
STATUS LINE 2 - is the status line containing the data and formula
COMMAND LINE - is the last line on the screen
WINDOW CURSOR - is the block within the spreadsheet which indicates the current location

In each case, enter the control sequence in hexadecimal format which controls the desired visual attribute. Multiple visual attributes (reverse video and half-intensity) are supported.

13 AUX port ON.....(in hex format)...

Enter the hexadecimal sequence which will activate the auxillary port. For example: "DC2" activates the aux port on most ADDS terminals. The hexadecimal representation of "DC2" is "12".

14 AUX port OFF.....(in hex format)...

Enter the hexadecimal sequence which will disable the auxillary port. For example: "DC4" disables the aux port on most ADDS terminals. The hexadecimal representation of "DC4" is "14".

15 Printer characters/line..(in decimal form).

Enter the printer width in characters per line which should be considered as standard for the port using these parameters. If no entry is made, 132 will be the default value. This value is entered in decimal form.

16 Printer lines/page.....(in decimal form).

Enter the printer depth in lines per page which should be considered as standard for the port using these parameters. If no entry is made, 62 will be the default value. This entry is made in decimal form.

17 Screen width-characters..(in decimal form).

Enter the width of your screen in characters. This value will be the standard setting and can be changed in COMPU-SHEET with the "/TERM" command. If no entry is made, 80 will be the default value. This entry is made in decimal form.

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

18 Go to command definition screen

This selection will display the command definition screen.

The command definition screen provides for equating any available key on the keyboard, such as function keys, with a COMPU-SHEET command. This feature allows COMPU-SHEET to be tailored to meet the specific needs and preferences of the user.

The command definition screen provides for the assignment of up to 60 keys on the keyboard. All that is required is the hexadecimal sequence generated by the key and the COMPU-SHEET command to be assigned to the key. For example: function key 1 is to be assigned to the "/PRINT" command. If function key 1 generates an "ESC:1" sequence, the "HEX VALUE" would be "1B31" and the "COMMAND" would be "/PRINT".

	Hex value	Command
1	1B31	/PRINT

Another example might be to assign the "P" key to the "/PRINT" command. In this example the "HEX VALUE" would be "50".

	Hex value	Command
1	50	/PRINT

This table can be used to tailor the features of any terminal to COMPU-SHEET.

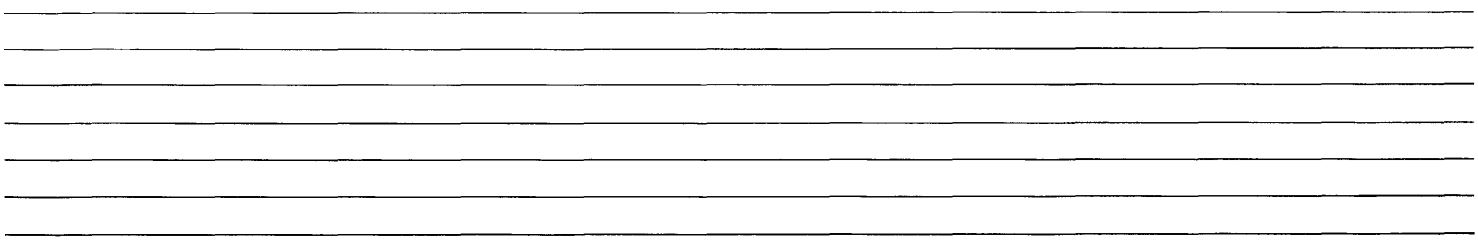
(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved

(C) Copyright 1984 Raymond-Wayne Corporation -- All Rights Reserved



GENERAL AUTOMATION