

WORD PROCESSING

T/MakersTM

ELECTRONIC
SPREADSHEET

QUICK REFERENCE BOOKLET

LIST
PROCESSING

GRAPHICS

DATA
TRANSFER

FILES
MANAGEMENT

Copyright © 1982 by T/MAKER Company
All rights reserved.

No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual, or otherwise, without the prior written permission of Peter Roizen, T/MAKER Company.

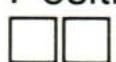
DISCLAIMER

No representations or warranties are made with respect to the contents hereof. Further, T/MAKER Company reserves the right to revise this publication and to make changes from time to time in the content hereof without obligation to notify any person of such revision.

useful terms

Working File—refers to all lines in the file

Position 1



Position 2 etc.

Line 1

Line 2

etc.

Working Area

Refers to all lines from the top line on the screen forward—the area normally affected by a command

Top line on the screen

Top line in the frame
(when in Frame Mode)

15/55 WHAT NEXT? **sort n 1 10**

Parameters or Options
(some commands require them)

Command name

Universal Prompt

Leftmost position shown on the screen (or in the frame)

Line number of the top line of the screen (or in the frame)



edit

To enter the editor:

WHAT NEXT? **edit** or **e**

To quit the editor: **CTRL Q**

When the editor is quit, the line on which the cursor rests becomes the top line on the screen (or the top line inside the frame, when in Frame Mode).

Revise Text

Insert character **CTRL W** _____

Insert line **CTRL R** _____

Delete character **CTRL E** _____

Delete line **CTRL T** _____

Delete spaces
to next character .. **CTRL U** _____

Destructive
backspace **DEL** _____

Erase to end of line .. **CTRL Y** _____

Join lines **ESC <** _____

Break lines **ESC >** _____

Move Lines

Clear the buffer **CTRL C** _____

Copy to the buffer ... **CTRL F** _____

Move to the buffer ... **CTRL G** _____

Insert the buffer **CTRL D** _____

Search and Replace

Search for string **ESC *** _____

Continue search **ESC &** _____

Replace strings **ESC ..** _____

Change Modes

Carriage return mode **ESC R** _____

Insert mode **ESC I** _____

Frame mode **ESC F** _____

Move Columns

Set the left limit **ESC (** _____

Set the right limit **ESC)** _____

Copy columns **CTRL S** _____

Move columns **CTRL V** _____

Tabs

Tab **TAB** _____

Set tab **ESC TAB** _____

Clear tab **ESC ESC TAB** _____

Set all tabs **ESC S TAB** _____

Clear all tabs ... **ESC C TAB** _____

Keystroke Macro

Begin macro **CTRL B** _____

Home macro **CTRL N** _____

End macro or
execute once **CTRL X** _____

End macro or execute
many times **CTRL Z** _____

Others

Quit the editor **CTRL Q** _____

Repeat last keystrokes **CTRL A** _____

Information **ESC ?** _____

Redraw the screen ... **ESC #** _____

(See "Addenda" for
IBM Users' Edit pages)

How to Move the Cursor

■ **ESC** **CTRL** **O** _____

Home the screen

■ **ESC** **A** _____

Move to first line

■ **ESC** **CTRL** **K** _____

Screen up

■ **ESC** **CTRL** **H** _____

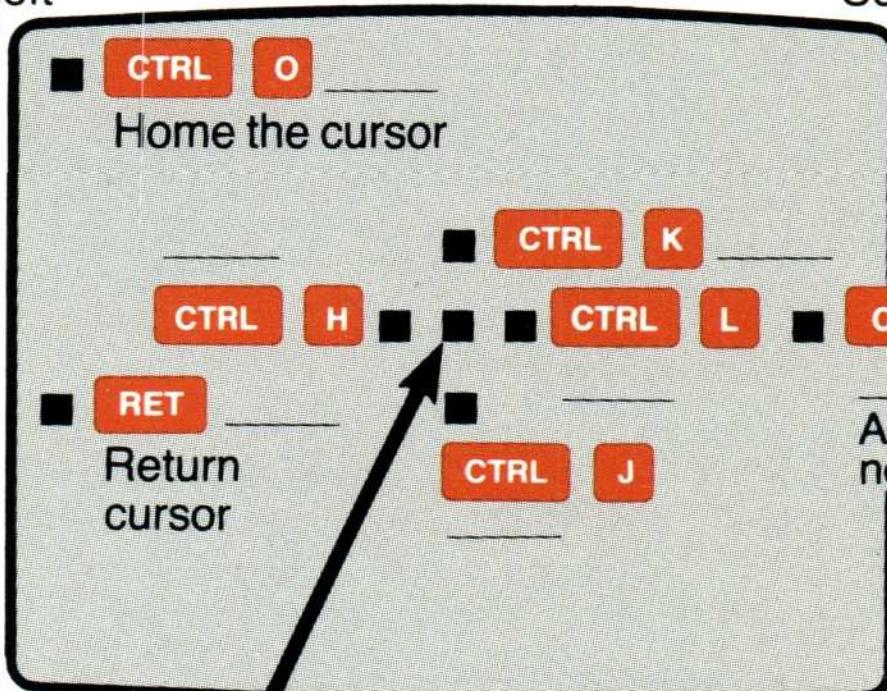
Screen left

■ **ESC** **CTRL** **L** _____

Screen right

■ **ESC**
RET _____

Return screen



You are here

■ **ESC** **CTRL** **J** _____

Screen down

■ **ESC**
CTRL **P** _____

Advance to
end
of line

■ **CTRL** **I** _____

Advance to next word

Tabs and Searching
also move the cursor.

■ **ESC** **Z** _____

Move to bottom line

How to Use the Keys

RET

touch Return Key

ESC

touch Escape Key

CTRL **X**

hold down Control

Key and touch letter shown

TAB

touch Tab Key or

CTRL **I** if the terminal
has no Tab key

DEL

touch Delete Key
(sometimes marked "Rub")

All keystrokes for editing commands
may be changed to the user's own prefer-
ence. See "Customization Notes" in the
Reference Manual.

align

To align the text in the working area:

WHAT NEXT? **align**

The text will be aligned according to aligning wedges placed in the file. Each set of wedges applies to the text that falls after it in the file until a new set of wedges appears.

To change the margins on one side, specify only the wedges on that side.

Paragraph Separation

Paragraphs must be separated by one of the following:

- a blank line
- a line containing aligning wedges
- a line with a period in position one (i.e., a “design command” line)
- a line with three or more dashes in succession
- a line with three or more underscores in succession

Tagged and Recalled Wedges

A pair of aligning wedges may be “tagged” by a one-letter name.

<< a >>

Tagged in this way, the wedges may be recalled for use later on in the file by inscribing a left-side pair of wedges, an equal sign, and the one-letter name.

<<=a

Aligning Options

Unaligned Text

>> <<

All work
and no play
makes
Jack a dull boy

<<<

>>

All work and no play makes
Jack a dull boy. All work and
no play makes Jack a dull
boy. All work and no play
makes Jack a dull boy.

<<<

->>

All work and no play makes
Jack a dull boy. All work and
no play makes Jack a dull
boy. All work and no play
makes Jack a dull boy.

JUSTIFIED

RIGHT-RAGGED

Centered Text

<<-

>>

All work
and no play
makes
Jack a dull boy

<<

>>

All work and no play
makes Jack a dull boy. All
work and no play makes
Jack a dull boy. All work and
no play makes Jack a dull
boy.

JUSTIFIED

RIGHT-RAGGED

print

To print the working area or a file on disk:

WHAT NEXT? **print** [options, if any]

[**it** or disk-file-name]

Printing in progress can always be terminated by typing **CTRL Q**

Options

Options, when used, are interposed between the word “print” and the designation of the file to be printed.

- nonstop** Pre-answers Printer’s Prompts with **Y**.
- number 5** The first page in the file is assumed to be page number 5.
- from 7** Prints from page 7.
- to 10** Prints through page 10.
- delay 9** Sets the speed of printing to the screen.

Bold and Underscored Text

A string of text can be made bold by preceding and following it with a vertical bar character (**|**).

A string of text can be underscored by preceding and following it with a back-slash character (****).

The characters used to underscore or make text bold occupy a character space but are not printed.

Note: See Design Commands for other ways to bold or underscore a complete line.

Note: Characters for bold and underscoring may be changed to the user’s own preference.

See “Customization Notes” in the Reference Manual.

The Printer’s Prompt

The Print command brings the “Printer’s Prompt” to the screen.

NEXT PAGE 1 (YES.SCREEN.NO.GO.QUIT)?

One of the following answers should be given:

Answer	Result
Y	Prints the page on paper.
S	Prints the page on the screen. The printing may be stopped temporarily (and re-started) by touching the spacebar.
N	Skips the page. The Printer’s Prompt appears for the next page.
G	Pre-answers all following Printer’s Prompts with y .
Q	Terminates printing session.

Design Commands

Design Commands carry out a number of editing functions. They are inscribed into the file but are not printed.

Design Commands always begin with a period in position one. Keywords in Design Commands (e.g., **single**) may be abbreviated to their first three letters (**sin**).

Note: See following pages for descriptions of Design Commands.

Design Commands

.width 80 Sets page or panel width (e.g., 80 positions).

.pagesize 66 Sets physical size of a page (e.g., 66 lines).

.length 60 Sets usable length of a page (e.g., 60 lines).

.number 3 Sets page number of current page (e.g., 3).

Note: **Number 0** sets page number of current page and following pages to zero until reset by another **.number** command.

.top 5

“Top Routine”—a series of lines to be printed single-spaced at the top of each page starting with the page indicated (e.g., page 5).

Only the following Design Commands may appear within a Top Routine:

.+ . . = . & . .. ,

Number signs (#) within Top Routines provide automatic page numbering.

.end

.block

Defines a block of lines to be printed on a single page.

Any of the following Design Commands will end the block:

.end	.block	.top	.bottom
.length	.pagesize	.newpage	.footnote

.end

.footnote

Defines a group of lines to be printed single-spaced at bottom of this page or next.

Footnotes are placed above Bottom Routines.

Only the following Design Commands may appear within a Footnote:

.+ . . = . & . .. ,

.end

.bottom 2

“Bottom Routine”—a series of lines to be printed single-spaced at the bottom of each page starting with the page indicated (e.g., page 2).

Only the following Design Commands may appear within a Bottom Routine:

.+ . . = . & . .. ,

Number signs (#) within Bottom Routines provide automatic page numbering.

.end

Design Commands

- . **single** Single spaces the following text.
- . **double** Double spaces the following text.
- . **wedges** Prints aligning wedges.
- . **nowedges** Terminates printing of wedges.
- . **clean** “Cleans” the following table before printing.
- . **noclean** Prints the following table as is.
- . **indent 6** Shifts the following text 6 positions to the right.
- . **indent 0** Prints the following text without indenting.
- . + Bold all previous line.
- . – Underscores all previous line.
- . = Underscores all words in previous line.
- . & Overprints previous line with next line.
- .
- . **Hi Jack!** Comment Line—text appears in file but does not print.
- . , Nonaligning Line—text printed unaligned, minus dot and comma.
- . 23 Advances printer to the 23rd line of the page.
- . **newpage** Advances printer to a new page.
- . **newpage 5** Advances printer to a new page if less than 5 lines are left on the current page.
- . **continue xxx** Printing automatically proceeds to disk file named (e.g., xxx).
The working file can contain numerous .**continue** commands. A file printed from disk can contain only one, and that must be on its bottom line.

compute

The *Compute* command carries out mathematical calculations on the working area.

WHAT NEXT? **compute** [c c c... as needed—used for carrying values from the bottom of a table back to the top]

All mathematical calculations are first laid out on the screen in table form (while in the editor) and then calculated by invoking the *Compute* command. This table must include an “example line” and numbers must be properly located beneath it. An optional “zero values” line may be specified. The first seven positions of each line are reserved for special purposes and may not be used for headings or numbers.

Example Line

The example line locates the zones in a table that will be used for columns of numbers and provides models of how those numbers should be presented (“model numbers”).

- **ex** in positions 1 and 2 followed by model numbers after position 7
- ex 9,999 99.9 ,99
- May contain up to 25 model numbers
- Can be installed at any point in table in order to change features of presentation

Placement of Numbers in Table

- Only within left and right limits set by model numbers in example line
- Must be contiguous (no blank spaces) including minus signs.
- Will be automatically aligned when entered correctly.

	Right	Wrong
ex	999,999.99	999,999
+	12.3	— 45
+	12,678.9	5678
+	-2156	1,234

Model Numbers

- Separated from each other by one or more spaces
- Consist of strings of nines (or other characters), optional commas, optional decimal point
- May contain up to 13 nines
- Will not print numbers below it when started by a comma (third column of example line above)

Zero Values Line

Though the example line specifies how numbers should be presented, it is often desirable that null values are given special treatment. The Zero Values Line provides this option.

- Follows example line
- **zv** in positions 1 and 2 followed by symbols designating how null values are to be presented

ex	999.99	999.99	9.9999
zv	—		none

Note: Values falling close to, but not exactly upon, zero are always presented in the fashion prescribed by the model nines, even when all of the digits shown are zeros.

compute

General Calculations

General calculations may be laid out horizontally (across columns), vertically (down rows), or across tables (by employing the *Combine* command). They consist of mathematical operators followed by a single terminator symbol placed in relation to a row or column.

- **mathematical operators**—define the operation to be performed upon the values to which they refer
 - addition (+) multiplication (*)
 - subtraction (–) division (/)
- **terminator symbols**—call for a result to be produced and printed. An equal sign (=) produces the following calculating sequence:

- (1) *Compute* begins with zero,

- (2) sums all numbers associated with plus signs (+), if any;
- (3) subtracts all numbers with minus signs (–), if any;
- (4) multiplies by all numbers associated with multiplication signs (*), if any;
- (5) and, finally, divides by all numbers associated with division signs (/), if any.

As an equation, *Compute* would be:

$$\frac{[(\text{Sum of } +\text{'s}) - (\text{Sum of } -\text{'s})] \times (\text{Product of } *\text{'s})}{(\text{Product of } /\text{'s})}$$

Note: If a general calculation contains no plus or minus signs, the result must be zero.

Terminator Symbols

A general calculation may be terminated by an equal sign (which generates the simple result) or by one of the other symbols listed below. Each of these additional terminator symbols can be thought of as performing one last calculation upon the simple result before generating an output figure.

Symbol	Modification
=	simple result
%	result multiplied by 100
#	result divided by 100
sfh	result subtracted from 100
sfo	result subtracted from 1
rec	reciprocal of result
sqr	square root of result
abs	absolute value of result
exp	e to power of result
log	natural log of result
l10	log to base 10 of result
sin	sine of result (radians assumed)
cos	cosine of result (radians assumed)
tan	tangent of result (radians assumed)
atn	arctangent of result (radians produced)

sgn	sign of result (+1, 0, –1)
ply	maximum of zero and result (positives only)
nly	maximum of negative of result and zero (negatives only)
avr	result divided by number of plus signs in equation (if equation only has plus signs, result would be the average)

Special Terminators

With special terminator symbols, + and *, are used to flag or identify values. The result is reported at the spot indicated by the placement of the special terminator symbol.

+ * max	selects maximum of two numbers identified by the + and *
+ * min	selects minimum of two numbers identified by the + and *
+ * ~	produces percent change in going from value identified by + to value identified by *
+ * ^	produces value identified by + symbol raised to power of the value identified by *
+ + cnt	produces count of plus signs in equation

Note: For three-letter terminators, the position of the first letter determines the appropriate strip or column.

compute

Vertical Calculation Strips

Individual calculation equations laid out in a single position are called Strips

- may only employ General Calculations
- must be located in positions 1-7.
- consist of one or more mathematical operators placed above terminator symbol
- can be linked together
- cause the calculation to be carried out on all columns of the table

Once a terminator symbol has been used in a Strip, a new calculation may be started below it in the same Strip. If more than one Calculation Strip terminates on the same row, the output result will be the sum of each Strip's result.

ex		999.99
+	Item A	40.00
+	Item B	60.00
= + +	Total Price	100.00
*	Percent Tax	5.00
= #	Price With Tax	105.00

(Figures in bold face are calculated)

Calculation Lines

Individual calculations laid out on single lines are called Horizontal Calculation Lines.

- may appear anywhere in a table
- may apply to the rows below
- always begin with a Control Code in positions 1 and 2 which specifies the set of rows upon which this calculation should be carried out.
- always have a Sequence Number next (0 to 20) which specifies the order in which this calculation will be carried out relative to other Calcula-

tion Lines. If a Calculation Line reuses a previously used Sequence Number, the old Calculation Line is replaced by the new.

- symbols must be placed in zones defined by the example line (see shaded areas)

When using General Calculations on a Calculation Line, the Terminator Symbol must be placed to the right of the other symbols. After a General Calculation has been terminated, a new one may be started on the same line. The order of such calculations is left to right.

	List Price	Price Paid	Percent Discount
ex	999	9999	999
ac1	/	+	%+ sfh
+	50	40	20

Calculation Line Control Codes

- ac** *Always Compute*—applies to all subsequent rows
- jc** *Just Compute*—applies only to the next row
- uc** *Usually Compute*—applies to all subsequent rows except those derived from other rows
- rc** *Rarely Compute*—applies only to subsequent rows derived from other rows
- cc** *Cancel Compute*—all previous Calculation Lines no longer apply below. To cancel the application of a single Calculation Line, follow **cc** with the Sequence Number of the Calculation Line to be cancelled
- sc** *Suspend Compute*—all previous Calculation Lines will not apply to next row. To suspend a single Calculation Line, follow the code with the Sequence Number of the Calculation Line to be suspended

compute

Restricted Calculations

Restricted Calculations may be used only in computations laid out horizontally, i.e., only upon Horizontal Calculation Lines.

Placing Constants into a Calculation

Constant values can be placed into a calculation by employing a special type of Calculation Line. This line consists of a calculation Control Code followed by a Sequence Number followed by a Special Symbol designating what is to be done with the constants entered into this Calculation Line. Different constants can be entered into the same Calculation Line. Columns left blank or presenting a zero value will be left unaffected by the constant computation.

ex	999	9999	9999	9999
uc1*		5	6	0

For example, in the case above, this Calculation Line would cause the value in the second column of the row to be multiplied by five and the value in the third column of the row to be multiplied by six. The values in the first and fourth columns of the row would be unaffected.

Special Symbols for Constants

- = column set equal to constant
- += constant added to value for column
- = constant subtracted from value for column
- *= value for column multiplied by constant
- /= value for column divided by constant
- %= value for column divided by constant and result multiplied by 100

- # value for column multiplied by constant and result divided by 100
- max** maximum of value for column and constant becomes new value for column
- min** minimum of value for column and constant becomes new value for column.
- ^** value for column raised to power of constant.

Store, Fetch, Pass, Round

The symbols described here may be given on a Calculation Line by themselves or they may be mixed with the symbols of General Calculations.

- st?** The interim value for the column is stored for future use. A single letter, in place of the question mark, becomes the stored value's "name."
- ft?** The stored value (a one-letter name in place of the question mark) is fetched and placed into the column associated with the site of **ft?**
- pas** The interim value for the column is "passed" to the next row. The value passed thus becomes the initial value for the same column in the next row. **pas** values will override other values for the initial value, e.g., values derived from a Vertical Calculation Strip.

- rnd** The interim value for the column is rounded to the accuracy of the model number that applies to it and calculations using this value will employ the rounded instead of the unrounded value.

Restricted calculations continued on next page

compute

Special Notations

A number of Special Notations may be used on Calculation Lines. These Special Notations must appear on a line by themselves. The symbols need not appear in the order shown below.

+ + + + mvr

The value of a + column is moved to the + column to its right.

+ + + + mvl

The value of a + column is moved to the + column to its left.

+ + + + cum

+ columns are converted to a cumulative distribution (left to right).

+ + + + chg

+ columns are converted to represent the difference (or change) between a + column and the column to its left.

+ + + + pch

+ columns are converted to represent the percent change between a + column and the + column to its left.

+ + + + pct

+ columns are converted to percents of the **pct** column.

+ + + + prj

The leftmost nonzero + column is projected to the right based on the growth rate in the **prj** column.

+ + + + inc

The leftmost nonzero + column is projected to the right based on the additions of the increment in the **inc** column.

- * * + grw

The growth rate in going from the – column to the + column is put in the **grw** column. * is used to mark intervening periods.

- / + grw

The growth rate in going from the – column to the + column is put in the **grw** column. The / column should contain the number of periods over which the growth has occurred.

dis + + net

The discount rate in the **dis** column is applied to the + columns to derive the net present value which is put in the **net** column.

The Order of Calculations

Tables are calculated one row at a time proceeding from the top to the bottom of the table.

For each row:

- If the row is derived from other rows,

the appropriate Calculation Strip is used to derive all columns of the row.

- The appropriate Calculation Lines as determined by Control Codes, are then applied in order as determined by Sequence Numbers.

an example

ex	9999	9999
ac 2 Calculation Line 2 (applies to all rows)		
uc 3 Calculation Line 3 (applies to data rows)		
rc 4 Calculation Line 4 (applies to derived rows)		
+	Row A (Data)	
+	Row B (Data)	
=	Row C (Derived)	
jc 1 Calculation Line 1 (applies to next row)		
*	Row D (Data)	
=	Row E (Derived)	

The diagram illustrates the sequence of calculations for each row. Row A (gold) is the starting point. Row B (teal) follows, with 'uc 3' applying to its first column. Row C (yellow-green) is derived from Row B using 'rc 4'. Row D (orange) is derived from Row C using 'jc 1'. Finally, Row E (white) is derived from Row D. Arrows show the flow of data from one row's calculations to the next row's starting point.

compute

A Complete Example

	BUDGET FORECAST				April through December	Year Total
	January	February	March			
ex	999,999	999,999	999,999	999,999	999,999	999,999
zv		—	—	—	—	—
uc1				sta	fta	
uc2*					9	
uc3	+	+	+	+	=	
+ Mortgage Payment	456	456	456	4,104	5,472	
+ Car Payment	340	340	—	—	680	
+ Other	600	625	650	5,850	7,725	
= - + Total Expenses	1,396	1,421	1,106	9,954	13,877	
+ Regular Salary	1,700	1,850	1,850	16,650	22,050	
+ Consulting	350	400	450	4,050	5,250	
= + / Total Income	2,050	2,250	2,300	20,700	27,300	
Expenses as						
% Percent of Income	68	63	48	48	51	
+ = Net Income	654	829	1,194	10,746	13,423	
jc4	cum +	+	+	+		
= Cumulative	654	1,483	2,677	13,423	13,423	

(Figures in bold are calculated.)

Explanation for the Order of Calculations

- Row A is processed by applying Calculation Lines 2 and 3.
- Row B is processed by applying Calculation Lines 2 and 3.
- a. Row C is derived from rows A and B according to the Calculation Strip.
- b. Row C is processed by applying Calculation Lines 2 and 4.
- Row D is processed by applying Calculation Lines 1, 2, and 3.
- a. Row E is derived from rows C and D according to the Calculation Strip.
- b. Row E is processed by applying Calculation Lines 2 and 4.

Explanation for a Complete Example

The first three Calculation Lines apply only to data rows because their Control Code is given as *usually compute*. The first Calculation Line sets the fourth

column equal to the third (see Store and Fetch). The second Calculation Line multiplies the value on the fourth column by nine thus making the estimate for the fourth column equal to nine times the March figure (see Placing Constants). The third Calculation Line sums the first four columns into the fifth.

Various Calculation Strips are used to derive rows of interest. The first three Calculation Lines apply to none of these derived rows. Just ahead of the last row, however, a Calculation Line has been added which will apply to the last row. It converts column values to their cumulatives (see Special Notations). Thus, after the Calculation Strip sets the Cumulative Row to the same figures as the Net Income row, Calculation Line 4 modifies the values in the first four columns to achieve the cumulative.

clean

The *Clean* command removes from a table in the working area the example line, the zero values line, the column names line (used by the Bar command), calculation lines and strips preparing the table for final printing. Print design commands are unaffected by the *Clean* command.

WHAT NEXT? **clean**

Notes

1. Design commands placed within a table tend to muddle its appearance. One tip is to disguise design commands as ordinary lines of text by beginning the commands in position eight instead of position one. After *Clean* removes the first seven positions, the formerly-disguised commands become legitimate design commands.
2. The **. clean** print design command provides another, and perhaps more powerful, approach to cleaning.
3. *Clean* never removes more than the first seven positions of a row even when a terminator symbol extends beyond position seven (e.g., **avr** with the **a** in position six). If such a situation is unavoidable, an easy solution is to start the text of the table in position 15 and use *Clean* twice in succession.

An Example of Clean

Before

Profit Analysis				
	Item	Cost	Sales	Profit
.end				
ex		999,999.99	999,999.99	999,999.99
ac1		-	+	=
+	Windows	125,167.80	242,234.50	117,066.70
+	Mirrors	66,145.10	110,499.10	44,354.00
+	Storm Doors	48,091.23	57,120.11	9,028.88
.block				
=	Total	239,404.13	409,853.71	170,449.58
.end				

After

Profit Analysis				
	Item	Cost	Sales	Profit
.end				
Windows	125,167.80	242,234.50	117,066.70	
Mirrors	66,145.10	110,499.10	44,354.00	
Storm Doors	48,091.23	57,120.11	9,028.88	
.block				
Total	239,404.13	409,853.71	170,449.58	
.end				

combine

The *Combine* command combines a table on disk with a table in the working area, generating a new table in the working area.

WHAT NEXT? **combine** *disk-file-name disk-file-symbols
working-area-symbols*

- Each parameter of the *Combine* command is separated by a space, but no spaces should appear within a parameter.
- The combined table's numbers are presented according to the model numbers provided by the working area's example line.

Specifying the Calculation

Following the *Combine* command, the user must specify three parameters.

1. The name of the disk file with which the working area table is to be combined.
(The second and third are General Calculation symbols telling how the two tables should be combined.)
2. The mathematical operator(s) which determines what is to be done with a cell value in the disk file table.
3. a. The mathematical operator(s) which determines what is to be done with the cell-value of the working-area table.
b. A terminator symbol specifying the result to be produced. This result becomes the cell-value of the newly generated table in the working area.

The General Calculation is applied to the cells of the disk file and the working area on a corresponding cell-by-cell basis.

Examples

To sum the numbers in the working area with those in a disk file named **Sales.mar**:

combine sales.mar + + =

To average the numbers in the working area with those in a disk file named **tab.2**:

combine tab.2 + +avr

To take the numbers in the working file as a percent of the numbers in a disk file named **actual**:

combine actual / +%

To make the table in memory equal to the sum of disk files named **jan**, **feb**, and **mar**:

combine jan + =
combine feb + + =
combine mar + + =

Noncongruent Tables

Ordinarily, *Combine* is used on tables having the same numbers of rows and columns. If, however, one table has fewer columns or fewer rows, values will be combined only where corresponding values exist. Where the calculation is not applied, the working area's values will remain unchanged.

arrange/match

The *Arrange* command reorders the sequence of columns for lines in the working area.

WHAT NEXT? **arrange** [/left-position right-position, etc] **end**

The columns to be reordered are identified by pairs of numbers defining their present boundaries. The first number marks the left boundary and the second number marks the right boundary of a column to be moved.

The new order of columns is determined by the sequence of number-pairs

following the *Arrange* command: e.g., in the case shown below, the column defined by the number-pair 30-35 will come first occupying positions 1-6. Columns which are not described by a number-pair will be lost.

Note: Columns of blanks may be moved around to provide white space in a table. Design commands are unaffected by *Arrange*.

An Example of Arrange—**arrange 30 35 25 28 10 19 end**

BEFORE							AFTER								
1	5	10	15	20	25	30	35	1	5	10	15	20	25	30	35
Male		Single			17	Ray		Ray	17	Single					
Male		Married			42	Bob		Bob	42	Married					

The *Match* command contrasts the working-area with a file on disk.

WHAT NEXT? **match disk-file-name**

This command prepares a summary of the differences between the two files which then becomes the working area.

The summary shows the origin of unmatched lines as **W>** for a line in the working area not on the disk file or **D>** for a line in the disk file not present in the

working area. The line numbers of unmatched lines also are shown.

Blank lines are ignored throughout and no attempt is made to match them. The user is responsible for contrasting the appropriate lines.

An Example of Match

THE DISK FILE

Table of Weights

1980 1981 1982

John	189	180	175
Mary	115	120	133
Bob	157	158	157

THE WORKING AREA

Table of Weights

1980 1981 1982

Mary	115	120	133
Bob	157	158	157
Alice	8	25	34

THE NEW WORKING AREA (Summary)

5 D> John 189 180 175
7 W> Alice 8 25 34

find

To advance the top line on the screen (or in the frame) to the first line containing a particular string:

WHAT NEXT? **find** string

replace

To replace all occurrences of a first string with a second string in the working area:

WHAT NEXT?
replace string1 string2

drop

To drop all lines in the working area that contain a particular string:

WHAT NEXT? **drop** string

keep

To keep only those lines in the working area that contain a particular string:

WHAT NEXT? **keep** string

How to Specify a String

A string is any series of characters. If a string contains no blanks and does not begin with a single or double quotation mark, it can be specified simply by typing it unaltered.

Examples: alphabet
4,5
Jone's
&#!\$'\$#.##'' ''

If a string contains a blank or starts with a single or double quotation mark, it must itself be enclosed in quotation marks. The enclosing quotation marks (single or double) must not be present anywhere in the string.

Examples: 'NY NY'
'He said',
"s"

sort

The Sort command organizes the elements of a list alphabetically or numerically in either ascending or descending order. The command sorts lines based on the zone identified by the left-position right-position parameters.

WHAT NEXT? **sort [options, if any]
left-position right-position**

Options

(may select one or none from each category—maximum of three)

When options are not designated, the defaults are: **a c all**

a sorts in ascending order
d sorts in descending order
c sorts characters alphabetically
n sorts numbers
all sorts all lines in working area
s *some*—sorts only within blocks
of consecutive lines that have a
plus sign in the first position (use-
ful for tables)

p *partial*—sorts only within blocks
of consecutive lines that have
some entry in the zone used as
the basis for the sort (useful for
subdivided lists)

Example Command: **sort n d 10 20**

Note: To sort on multiple zones,
sort first on the least important zone, then
the next most important, etc.

Examples of Options

The lines within unshaded areas would be sorted by the option chosen.

ALL

John	31
Bob	21
Bill	52
Mary	25
Jane	26
Rose	51

SOME

ex	Name	Age
+	John	31
+	Bob	21
+	Bill	52
= +	Total	104
+	Mary	25
+	Jane	26
+	Rose	51
= +	Total	102
	= Grand Total	206

PARTIAL

Name	Age
John	31
Bob	21
Bill	52
Mary	25
Jane	26
Rose	51

tally

Tally counts instances of like elements on a list and presents a summary tabulation.

WHAT NEXT? **tally** [*it or disk-file name*] [**all**]
[*key-left-position key-right-position, etc.*]
[**sum** *left-position right-position, etc.*] **end**

Options:

- tally the working file (**it**) or a disk-file (*disk-file name*)
- tally the column (key zone) of one's choice by specifying the position-range of the column to be tabulated
- tally subgroups within groups by specifying multiple key zones.
- sum the values of additional variables by indicating **sum** and the position boundaries of the columns to be summed

Tally normally considers only lines that begin with a plus sign in the first position. If the **all** option is used, then all nonblank lines are tallied. The Tally command is terminated by the word **end**.

The Report

Tally prepares a report which is inserted into the working area. This report has the form of a data file. It can be saved and used directly with the Load command and appropriate masks to prepare summary tables.

The report shows the command used to generate it and the resulting tallies. A result line shows first the values found in the key zones specified. When more than one key zone is used, the values are separated by colons. This is followed by an equal sign and a number representing how many times the combination occurred. Finally, the group totals for any summed variables are presented in the order requested.

Example commands:

tally it all 1 5 end tally survey 1 10 sum 14 20 end tally it 1 10 20 25 end

An Example

Result of: **tally list 8 14 29 36
sum 41 47 19 22 end**

Disk file "LIST"

Sex	Age	Status	Income
8-14	19-22	29-36	41-47
+ Male	29	Single	28,000
+ Female	22	Single	19,000
+ Male	31	Married	22,000
+ Female	42	Married	36,000
+ Male	51	Divorced	42,000
+ Male	20	Single	15,000
+ Female	25	Single	12,000
+ Male	34	Married	18,000

TALLY LIST 8 14 29 36 SUM 41 47 19 22 END

Male	:	Single	=	2	43000	49
Female	:	Married	=	1	36000	42
Male	:	Married	=	2	40000	65
Female	:	Single	=	2	31000	47
Male	:	Divorced	=	1	42000	51

bar

Bar prepares a bar chart based upon the values in either one column or one row of a table. This table must be the working file and must also conform to the format requirements of the *Compute* command. (See below for options.)

WHAT NEXT? **bar [column-or-row-name or it] [options] end**

The user can choose how the bar chart will be presented by experimenting with various options. Each variation can be examined by specifying one or another option and pressing the carriage return key. The **end** should not be typed until a return to the Universal Prompt is desired.

Bar charts may either be displayed or placed into the working file for editing and printing.

Designating the Row or Column to Be Charted

The row or column of values to be charted is designated by name. In most cases, it is sufficient to provide the name appearing at the top of a column or the name at the side of a row.

A column name will cause values in that column to be rendered into bars; a row name will cause values in that row to be rendered into bars. Names containing a blank space must be enclosed by quotation marks and must appear exactly as they do in the table. The name **it** may be used to chart the row of values at the top of the screen (or top of the frame).

Specifying Values to Be Charted

The options below control which values will be charted in a given row or column. When neither option is specified, the default is **some**.

some When a column of values is to be charted, this option will include only those rows with + in the first position.

When a row of values is to be charted, this option will exclude columns whose model numbers begin with a zero. All other columns will be included in the chart.

all All values in the column or row are charted.

Overriding the Row or Column Designation

Normally, the location of the name (at the top of a column or at the side of a row) will determine whether a row or a column of values is to be charted. This can be overridden by the following options:

row forces *Bar* to use a row of values

col forces *Bar* to use a column of values

Examples of the Values Charted by Various Commands

bar Supplies all	bar Increase
bar Total	bar 1979 all

Expenses

ex	999,999	999,999	0,999,999
cn	1978	1979	Increase
ac1	-	+	=
+ Consultants	42,521	51,685	9,164
+ Supplies	11,256	18,123	6,867
+ Rent	8,200	10,600	2,400
+ Other	17,235	15,678	-1,557
Total	83,057	100,599	16,874

Column Names

A line beginning with **cn** may be included in a table to specify names to be used for the columns when they are presented in a bar chart. (See example above.)

Presentation Options

Presentation Options control various features of the bar chart's appearance.

numbers	causes numbers identifying bars to be placed in a column down the left side of the chart
nonumbers	returns the bars to an unnumbered condition
names 11	provides 11 positions for the names of bars (up to 100 positions may be requested)
names 0	suppresses the names of the bars altogether
values 6	provides six positions for the values corresponding to bars (up to 100 may be requested)
values 0	suppresses showing the values
	min -2250 sets the lower limit of the scale
	len 50 sets the number of positions spanning the lower and upper limit
	max 10000 sets the upper limit of the scale
	nobars suppresses showing bars
	bars bars restored
	width 2 sets the width of bars to two lines
	space 1 sets the spacing between bars to one line
	char □ sets the character used to draw bars
	short shows only one screen of the bar chart
	long scrolls all bars onto the screen (the space bar maybe used to interrupt the scrolling)

Category	Value
Consultants	9,164
Supplies	6,867
Rent	2,400
Utilities	668
Other	-1,557

Disposition Options

Disposition Options control the bar chart's fate. Ordinarily these options should be used only when the *Bar* command has been invoked at the top of the working file.

- replace** replaces the working file with the current chart
- insert** inserts the current chart ahead of the first line in the working file
- stack** stacks the bars of the current chart onto the end of the bars of the chart previously inserted

restart deletes the charts ahead of the initial working file

cut deletes the original working file while keeping any inserted charts

Use the following options for reviewing the result of commands entered:

- file** displays the initial working file on the screen
- chart** displays the charts inserted ahead of the initial working file

load

Load fills out specific zones in the working area with either data from a disk file or with blanks.

WHAT NEXT? **load** [disk-file-name or blank]

In order to use this command, the working area must be a mask, and the disk file must be a data file.

Data Files

A data file consists of lines which assign values to names. To do this, the name of a value is placed on the left side of an equal sign and its value is placed on the right side.

- Lines without an equal sign in a data file are considered to be comments.
- Values containing a space must be enclosed in single or double quotation marks.

Though a line may contain only one name, it may contain more than one value. In such cases, the values must be separated from each other by one or more spaces. The names of successive values are automatically assumed to be "one higher" than the name of the value preceding them.

- For a name ending with an alphabetic character, "one higher" means appending "1" to the end of the name (e.g., **x45y** becomes **x45y1**).
- For a name ending with a number, "one higher" means adding one to that number (e.g., **gnp9** becomes **gnp10**).

Masks

A mask must show both where data should be placed and how data should be presented. To show where data should be placed, braces are used to define the correct zone on a line. To show how data should be presented, the left brace is immediately followed by one of the special characters below.

Character	Interpretation
-	Data centered in zone
>	Data right-justified with right-most limit of zone
<	Data left-justified with left-most limit of zone
!	Data left-justified and unused space after data eliminated ("squeezed")

A name, which corresponds to the name used in the data file, should be put inside the braces to show which data should be placed there. If no name is put inside the braces, the name will be assumed to be one higher than the previous name, just like data files.

(See also "Long Names in Masks")

An Example of Load

A Data File (on disk)

```
Address Information
company = 'Doe & Sons, Inc.'
name = 'John Doe'
street = '1431 West 45th Street'
city = 'New York'
state = NY
zip = 10028
Sales Figures
sales.80 = 20,000 21,000
```

A Mask (the working area)

```
Name: { <name >
Address: { <street >
           { <city >
             !state } { !zip >
           }
         -company >
         1980   1981
Sales:   {>sales.80} { }
```

The Result of Load (the new working area)

Name:	John Doe
Address:	1431 West 45th Street
	New York
	NY 10028
Doe & Sons, Inc.	
Sales:	1980 1981
	20,000 21,000

unload

Unload moves (or “unloads”) data from the working area to create a new working area which is a data file.

WHAT NEXT? ***unload disk-file-name***

The file on disk must be a mask. *Unload* converts the working area from a text file to a data file on the basis of the mask. The original working area is destroyed in the process.

Masks

When using *Unload*, a mask shows where data are located in the text file and the names of data items to be unloaded. Braces on a line in the mask mean that the corresponding line in the text file should be unloaded. If a line in the mask has no braces, it is ignored.

The lines in the mask must be in the same order as those in the text file. The mask line must match the text line, character for character, for all positions to the left of the first brace on the mask line. This brace can be either an open brace, which would signal the first value to be unloaded, or a close brace, which simply demarks the end of the matching zone.

Long Names in Masks

To enter a data name which is longer than the zone for the data:

{ } place the braces to define the correct zones for an entire line
{sa} type the data name for a zone inside the braces until the cursor is over the close brace character

enter insert mode (see Editing Keystrokes)
{sa#} type a number sign (#)
{sa#les} type the rest of the name
leave insert mode

An Example of Unload

A Mask File (on disk)

This mask unloads name and 1981 Sales

Name: {name }
Sales: } {sales.81 }

A Text File (the working area)

Name: John Doe

Address: 1431 West 45th Street
New York
NY 10028

Doe and Sons, Inc.

Sales: 1980 1981
 20,000 21,000

A Data File (the new working area)

name = 'John Doe'
sales.81 = '21,000'

files management commands

Function	WHAT NEXT?	Notes
clip To discard all lines of the working file before or after the top line on the screen.	<i>clip before or after</i>	"Before" and "after" may be abbreviated to their first letters.
create To create a working file with a given name.	<i>create file-name</i>	
data To establish a default disk drive for all T/Maker text files. Thereafter, any file name lacking a specified disk drive will be assumed to refer to the default drive. To return to no default drive, give a space as the drive name.	<i>data drive-name</i> (for example, a or b)	
	<i>data ' '</i>	
delete To delete a file on disk.	<i>delete disk-file-name</i>	
do To process the top line on the screen (or in the frame) as if it were typed in after the Universal Prompt.	<i>do</i>	This line is automatically deleted from the working file.
files To inventory the files on a disk drive.	<i>files drive-name</i>	The listing may be interrupted and restarted by touching the space bar.
get To make a file on disk the working file.	<i>get or g disk-file-name</i>	
info For information about the working file.	<i>info</i>	
insert To insert a file on disk at the top of the working area.	<i>insert disk-file-name</i>	
[line-number] [/position-number] To move the upper left-hand corner of the screen (or of the inside of the frame) without entering the Editor, type the desired coordinates of the new location. Only one coordinate need be specified if the other is not to be changed.	Examples: 25 and 35/40 and /150	

files management commands

Function	WHAT NEXT?	Notes
list To view a file on disk without affecting the working file.	list <i>disk-file-name</i>	The listing can be interrupted and restarted by touching the spacebar. (See <i>Print</i> for printing on the screen.)
merge To superimpose a file on disk on top of the working area.	merge <i>disk-file-name</i> <i>position</i>	The position parameter defines the start of the horizontal zone of the working area to be overlaid.
notabs (see tabs) To stop subsequent Save commands from saving tabs.	notabs	
rename To give a new (and additional) name to the working file.	rename <i>new-file-name</i>	
reset To prevent a BDOS Read Only (R/O) Error, when attempting to alter a newly inserted disk, type after inserting the new disk.	reset	
save To store the working file on disk.	save or s	This command will save tab stops with the file if the <i>Tabs</i> command was previously invoked. If a previous version of the named file is on disk, that version is renamed <i>file-name.BAK</i> .
stop To stop T/Maker and return to the operating system. To interrupt most major functions in progress.	stop CRTL Q	
tabs (see notabs) To save tabs with subsequent Save commands.	tabs	
wait To suspend a series of commands until a key is touched at the console, add to series.	wait	

error messages

General Errors

Most error messages in T/Maker are self-explanatory. Below, a few common problems are described.

OUT OF SPACE

The working file is getting too big. Enter the editor and delete whatever you can or restructure the file into smaller pieces.

DISK FULL

The Save command was used and the disk or disk directory is full. Use the *Files* command to list what is on the disk, delete anything unnecessary, and then give the Save command again.

... NONEXISTENT

A command or file was called for which does not exist. Check the spelling of the command or file name.

... EXISTS

An attempt was made to use a file name that already exists. Pick another name or delete the existing version of the file and try again.

WRONG NUMBER

A value which was supposed to be a number is not a number or the number specified is out of the allowable range.

TOO BIG

TOO MANY

TOO COMPLEX

TOO MUCH

The size of some file, problem, section, or whatever, is too great. Make smaller.

Mathematical Errors

A number printed as asterisks means the model number provided on the example line was not big enough for the value found in the table.

... MINOR ERRORS

This is a count of instances of too-large numbers and a variety of other mathematical errors including division by zero, taking the square root of a negative number, calculating a growth rate from zero, etc. If a calculation is in error, zero is produced as the result.

addenda

The following apply only to 16-bit versions
of T/Maker

General	A line may be up to 400 column-positions long instead of 300.
Files Command	May be used with ambiguous file names involving “ ” or “?”. Examples: files b: *.bak files a: ??83.tab
Compute Command	An Example Line may contain up to 50 Model Numbers instead of 25. Calculation Line Sequence Numbers may range from zero to 30 instead of zero to 20.

T/MAKER Company
2115 Landings Drive
Mountain View, CA 94043