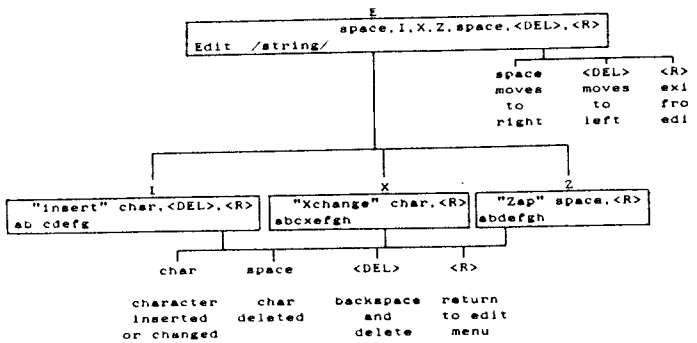
EDIT COMMAND



Entering the edit mode

You can edit the contents of any cell. While editing the automatic calculation feature will be switched off for expressions and mistakes in your logic will not be picked up. However on completion of the editing a full recalculation will be carried out.

You will not therefore know about any errors until you have left the edit mode, which is in direct contrast to what happens when you insert an expression for the first time. You may find it easier to overwrite expression entries rather than edit them, although no real harm can be done either way.

If an error is found when the recalculation is attempted the line in question will be displayed on the edit line up to the character in error. You can correct the line at this stage by using the [DEL] key to remove characters. Then make insertions just as if you are in the entry mode and finish off with a RETURN.

The EDIT command is entered by typing 'E' followed by a RETURN after which the relevant cell, the current cursor cell, will be displayed on the edit line and the cue will be located at first count.

Moving the cue

To move the cue to the right just press the space bar once for each character. To move the cue to the left press the [DEL] (or backspace if it works on your machine) key once for each movement.

Insert:

If I is typed at any particular location, the entry will split about that point creating a blank space into which a new character can be inserted. If you type something in a further space will then be created for the next character to be inserted. Pressing the RETURN key will cause removal from the insertion mode and the remaining space will be deleted.

eXchange:

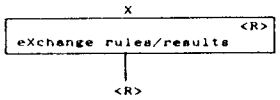
Typing X will allow the character at the current cue location to be replaced by the next entered character. The RETURN key may be used to leave this mode and go back to the main edit menu.

Zap:

Typing Z at any cue location will set you up for deletion. Every time you press the spacebar a character will be deleted. To get out of Zap mode and back to the main edit menu just press the RETURN key.

While not in Insertion, eXchange or Zap mode pressing RETURN will cause removal from the EDIT mode and the amended line will be placed into the relevant worksheet cell. Alternatively the arrow keys may be used to leave the editing of the cell and go into entry mode on an adjacent cell.

EXCHANGE COMMAND

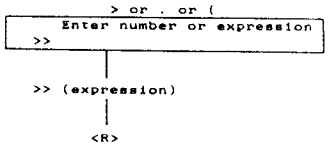


The eXchange command is entered by typing an X followed by a RETURN. The effect of this command is to change the display in the cells from the numerical results to the rules or formulae from which they are calculated. It will usually be necessary to change the column width with the New Width command sequence. Go back to display of the answers by typing X once more.

Together with the formulae each expression cell is displayed with its order of calculation number in brackets followed by its formula.

Note that this is not the same command as the eXchange used in EDIT mode.

EXPRESSION ENTRY



An expression is rather like a mathematical formula that can be entered into a given cell. The formula normally remains invisible but THE CRACKER will work out the resulting solution and display that in the cell when it is on screen.

An expression is typed in on the entry line (after Next:) and on completion is inserted into the memory together with an order of calculation number. On completion the memory remaining is also calculated and displayed and if automatic calculation is switched on the worksheet is recalculated.

As you type the value of the expression will be worked out and displayed in the target cell, but the expression itself will be displayed inside brackets on the cell contents line at the top.

Evaluation of expressions

Expressions may contain functions, coordinate references, numbers, arithmetic operators, logical operators and special purpose operators. Expressions are evaluated according the the rules of normal algebra.

As in algebra all operators are given an importance which determines the order in which the various parts of the expression are calculated. This order of importance is over-ridden by parentheses, each set of which is evaluated as if it were a self- contained expression. The innermost set of parentheses is calculated first and then the next innermost and so on working outwards.

In order to multiply two sets of parentheses an asterisk symbol must be put between them, which is the computing convention for a multiply sign. You cannot have (2)(3) for 2*3 instead you must put (2)*(3).

The operators within the expression or set of parentheses are calculated in the following order: 'plus and minus', 'percentage', 'exponentiation', 'multiplication and division', 'addition and subtraction', 'greater than' and 'less than' and 'equal', 'greater than or equal' and 'less than or equal' and 'not'. The symbols to use for each of these commands will be given later.

If an exclamation mark is encountered on the entry line then the value of the expression to that point is determined and replaces the whole of the expression on the edit line.

If, for example, you have typed an expression such as 2*B1+20 the resulting value displayed in the cell will change if the entry that is held in B1 ever changes. By inserting an exclamation mark thus 2*B1!+20 the expression is calculated using the value that is held in B1 at that time. The resulting entry is therefore constant and no longer dependent on B1.

It should be used with caution as indeterminate results may occur if for example it is used from within a function.

The valid parts of an expression are as follows: operators, coordinates, defined functions, numbers and conditionals. Functions comprise a named operation with its required parameters in brackets and where there are more than one these parameters must be separated by commas. There are three types of function, those whose arguments have multiple parameters, those which have single parameters and those which have no parameters at all.

When an apparent function name is encountered it is checked against the list of inbuilt functions. Functions may call other functions in their parameters. Indeed a function may even call itself amongst the parameters. As this is a stacked operation the number of levels of nesting is limited, however the program keeps a track of the state of the stack. There is no need to worry about these technical terms, all you need to know is that if the nesting has become too great (i.e. if there are too many levels of brackets in your expression) an error message will come up and it will only be necessary to rearrange the expression so that it is placed into two entries rather than one. The expression so far is not lost, it is merely necessary to use the [DEL] key to backspace.

Expressions are evaluated as you enter them, not on completion. This means that you have dynamic error checking character by character. If you enter a wrong character you will get an immediate error message. To correct an error, backspace with the [DEL] key and then enter a new character.

Note that it is impossible for THE CRACKER to check for errors that are 'legal' in the context of the program but are not the correct expression that you intended to enter.

If you are entering a complex expression such as a DO-WHILE formula then the calculation character by character can get bogged down. In this case first switch off the automatic calculation. See the AUTOMATIC CALCULATION command.

Arithmetic Operators

+ Addition
- Subtraction
* Multiplication
/ Division
^ Exponentiation (raising to a power)
% Percentage

Logical Operators

Logical Operators act on arithmetic values or expressions (A and B below) and determine whether they comply with specified conditions.

= Equal	(A = B)
> Greater than	(A > B)
< Less than	(A < B)
] Greater than or equal	(A] B) not all machines support this
[Less than or equal	(A [B) not all machines
| Not equal	(A | B)

If the expression in brackets is met then the result is TRUE and given the value of -1. If it is not met then it is FALSE and given the value 0. (Don't worry about why this should be so, it has its roots deep in the past of computer convention). See the Tuition section for more details.

(3.4] 1.23) = TRUE
(3.4 < 1.23) = FALSE

BUILT IN FUNCTIONS

Trigonometric Functions

SIN(X)
COS(X)
TAN(X)

Determine the sine, cosine and tangent respectively of X, where X is in degrees.

SINR(X)
COSR(X)
TANR(X)

Calculate the sine, cosine and tangent respectively of X, where X is in radians.

There is no restriction on the size of X as 2*PI (or 360 degs) will be repeatedly subtracted until X is within range. X may be negative.

ASIN(Y)
ACOS(Y)
ATAN(Y)

These functions determine the angles whose sine, cosine or tangent respectively is given by Y. The result is given in degrees in the range 90 degs to -90 degs.

ASINR(Y)
ACOSR(Y)
ATANR(Y)

These functions determine the angles whose sine, cosine or tangent respectively is given by Y. The result is given in the range PI/2 to -PI/2.

Y must be less than or equal to 1.

Logical Functions

Logical functions analyse a list of logical values or expressions (given as X,Y,Z... below) and return a value of TRUE or FALSE as defined above. In practice each expression or value is tested to see if it is -1 or 0. If it is -1 it is taken as TRUE, if it is 0 it is taken as FALSE.

AND(X,Y,Z....)

X,Y,Z.... are in turn checked for truth. If they are all TRUE then the function returns a value of TRUE (-1).

```
AND (TRUE,TRUE,TRUE)  = TRUE
AND (TRUE,FALSE,TRUE) = FALSE as list not all TRUE
```


OR(X,Y,Z....)

X,Y,Z.... are in turn checked for truth. If any of them are TRUE then the function returns a value of TRUE (-1)

```
OR(TRUE,FALSE,FALSE)  = TRUE one item is TRUE
OR(FALSE,FALSE,FALSE) = FALSE as none is TRUE
```

NOT(X)

The truth of X is checked and the opposite is returned as the function value. If the value of X is TRUE then FALSE (0) is returned. If the value is anything other than TRUE then TRUE(-1) is returned.

```
NOT(TRUE)  = FALSE
NOT(FALSE) = TRUE
```

UTILITY FUNCTIONS

ABS(X)

The absolute value of X is returned. It is defined as the numerical value of X with a positive sign.

```
ABS(-2.345) = 2.345
ABS(2.345)  = 2.345
```

AVERAGE(list)

The average of the non blank values in the list is calculated.

```
AVERAGE(3,4,5) = 4
```

BLEEP

Sets off a bleep. Use it as an audible signal.

CHOOSE(N,list)

The nearest integer value to N is found and the Nth item in the list is returned.

```
CHOOSE(3,5,6,7,8,9) = 7
```

COMB(N,R)

The number of ways of combining R items from a total of N is

calculated.

COUNT(list)

The list is checked for the number of values that are not blank.

COUNT(2,3,4,B5) = 3 : cell B5 is blank.

CRD(X,Y)

Can be used in any location where a coordinate is expected. The two arguments are column and line. The column must however be expressed in a numerical form (A=1, Z=26 etc). This function is useful in filling up tables from DO iterations.

DEC(X)

Returns the value of X minus 1. This is also intended for DO-WHILE iterations, primarily to act as a counter.

DEC(2.345) = 1.345

DO(range)

Allows subroutines, looping and iteration on the specified range. If iteration or looping is required the line will need to be terminated with a WHILE function.

DPART(X)

Takes the value of the decimal part of X, that is the part after the decimal point. Be warned that this function loses one significant figure for each figure that was previously before the decimal point. If you get unexpected results consider this.

DPART(5.78) = 0.78
DPART(-3.45) = -0.45

e

Gives the exponential constant 'e'

e=2.7182818284905

ERROR

A special function that if encountered in a calculation will bring up a message that an error has been called from that cell. Usually used

to check that values entered are in a permissible range.

EXP(X)

Raises the constant 'e' to the power in the bracket following. A check is made to ensure that the exponent is not too great.

EXP(3.14159265358979) = 23.1406926327793 exp(pi)

FACT(X)

Works out the factorial of X.

FACT(3) = 6

FALSE

Takes a value of FALSE, that is 0.

GROW

Will increase a value N by a percentage. The percentage can be negative.

GROW (5,200) = 210

IF(logic expr), THEN(expr 1), ELSE (expr 2)

This group is used to build decision making into the worksheet. The logical expression is analysed and if TRUE, THEN expression 1 is used to get the value for the cell. If the logical expression is FALSE then the ELSE expression 2 is used instead. The expressions themselves can be further IF... sequences building up extremely complex decision making sequences.

INC(X)

Returns the value of X plus 1. It is general purpose but is included for convenience in creating loops.

INC(2.345) = 3.345

INIT(crd,val)

Will initialize a cell within a specified value. The cell entry with this function will have a low recalculation number. The function is used for setting up loops and iterations. It is very similar to the SET function.

INT(X)

Takes the value of the integer part of X, that is the part before the decimal point, with the sign retained.

NOTE this definition as it may not be the same as that to which you are used if you are familiar with some computer languages such as BASIC.

```
INT(5.78) = 5

INT(-3.45) = -3
```

INTERP(N,range)

Compares N with each value in the range to find the two values between which N lies. The two values from the adjacent line or column are then interpolated to give the final value. The interpolation is done by taking the proportionate distance than N lies between the first two values and applying it to the adjacent values.

LOG10(X)

Determines the logarithm to the base 10 of the value in brackets.

```
LOG10(2.71828182845905) = 0.43429448190325  log10(e)
```

LN(X)

Determines the natural logarithm to the base 'e' of the value in brackets. The routine will check for negative or zero arguments which are not allowable.

```
LN(10) = 2.30258509299405  loge(10)
```

LOOKUP(N,range)

N is compared with each value in the range to find the first one it is greater than. The value from the adjacent line or column is then returned.

MAX(list)

The list is scanned and the maximum non blank value returned.

```
MAX(2,7,3) = 7
```


MIN(list)

The minimum non blank value in the list is returned.

```
MIN(2,7,3) = 2
```

NINT(X)

The nearest integer value to X is found.

```
NINT(2.23) = 2
NINT(5.67) = 6
NINT(-3.45) = -3
```

NPV(%,list)

Calculates the present value of the cash flow list using the discount % as specified.

PERM(N,R)

Finds the number of ways of permuting R items from a total of N.

PI

Returns the value of the constant PI.

```
PI = 3.14159265358979
```

POS(X)

Gives the value of the argument if it is positive. If it is negative then it returns a value of 0.

```
POS(2.345) = 2.345
POS(-2.345) = 0
```

RND

A true integer random number between 0 and 127 is found. This number is taken from the z80 refresh counter so the function is not available on computers with other processors.

```
RND = 23
RND = 120
RND = 3
```

ROUND(X)

Takes the value of X rounded to two decimal places. It is of use in financial calculations to avoid cumulative errors caused by including fractions of a penny (cent etc.). The rounded value is not just displayed it is also the value stored in the internal memory.

```
ROUND(1357.5679) = 1357.57
```

SET(crd,value)

Will set a cell to a particular value. The cell must initially be either blank or a constant. A formula will not be over-written. This command is usually used with the DO function to fill up a table with values.

SIMPRULE(step,range)

Works out the numerical integration of the range by Simpson's rule. The first parameter is the step length. Range holds the values for integration. The range must be an odd number of values consecutively. The latter parts of the range can be blank and these will be ignored. This makes flexibility in setting up worksheets possible.

SQRT(X)

Determines the square root of X.

```
SQRT(16) = 4
```

STDEV(list)

Looks through the list and works out the standard deviation. (See BYTE Nov 1983 pp560-563 for the algorithm used).

SUM(list)

Will add up the non blank values of the list and will return the total.

```
SUM(2,3,4) = 9
SUM(-3,4,5) = 6
```


SUMIF(crd1,list)

This is a conditional adding up function. Only those items in the range crd2...crd3 will be included that correspond to a cell entry in the column specified by crd1 and are on the same line.

	A	B
1		123.45
2	MARKER	23.45
3		245.56
4	MARKER	45.67

SUMIF(A1,B1...B4) gives 69.12
SUM(B1...B4) gives 438.13

NOTE crd1 can point to any cell in the column but must not point to a cell with a text entry in it or you will get an error message.

TRUE

Takes a value of TRUE, that is -1.

WHILE(expr)

Must be used together with a DO function. The expression must be logical and if TRUE the line will be recalculated from the start of the DO on that line. If FALSE the WHILE will do nothing.

VAL(X,Y)

Gives the numerical value of the cell specified by the two parameters. The coordinates are defined in the same way as for the CRD function.

VAR(list)

Looks through the list and works out the variance. The variance is the standard deviation squared.

DATE AND TIME FUNCTIONS

DATEAFTER(date,days)

Gives the date that will be the number of days specified after the input date. The date must be in the form of a single number, for example 312.1985 - being the 3rd December 1985. Be careful to put months 1 to 9 as 01 to 09.

DAYSAPART(date1,date2)
Gives the number of days between any two specified dates.

DAYOFWK(date)
Returns the day of week as a number. Saturday has a value 0, Sunday 1, Monday 2 etc.

DAYOFYR(date)
Returns the number of days between January 1st and the present day.

The rest of these functions are only available on CP/M PLUS (3) systems. Note for your clock and calendar to be correct you must set them before starting THE CRACKER. Use the utility called DATE, just type DATE SET and answer the questions.

YEAR
Gives the current year.

MONTH
The month.

DAY
The day

HOUR
The hour

MINUTE
The minute

SECOND
The second

DATE
Returns the date in the form of a single number, for example 312.1985 - being the 3rd December 1985.

ZEROTIME
Resets the elapsed time counter. Probably best included in an IF,THEN,ELSE entry.

TIMELAPSE
Returns the elapsed time since the ZEROTIME function was last operative. This is in seconds.

DELAY(n)
Does nothing until n seconds have elapsed. It may be a cell reference or a value. For practical reasons make it a cell reference with a value 0 until you actually want to run you application.

GRAPHICS FUNCTIONS

See the section on Graphics for a full explanation.

TYPEPLOT(crd)
The type of plot you want. Give a value between 1 and 6 in a cell reference to this value.

PLOT TYPES

Business:

1. Bar chart (histogram)
2. Stacked bar chart
3. line chart
4. Area chart
5. Hi-lo chart
6. Pie chart

Statistical, engineering and scientific:

7. X:Y line joining points
8. LogX:Y line joining points
9. X:LogY line joining points
10. LogX:LogY line joining points

11. X:Y points only
12. LogX:Y points only
13. X:LogY points only
10. LogX:LogY points only

15. X:Y points and best fit line
16. LogX:Y points and best fit line
17. X:LogY points and best fit line
18. LogX:LogY points and best fit line

MAINTITLE(crd)
The cell coordinate is a pointer to where the main title is to be found.

SUBTITLE(crd)
The same idea for the subtitle.

YTITLE(crd)
The pointer to the title up the Y axis on the left hand side.

XTITLE(crd)
Where the X axis title is. This is the one at the bottom.

XLABEL(crd...crd), or XLABEL(crd,crd,crd,etc.)
This points to the X labels which are the legend box items on the right.

Unlike the other functions in this list the range can also be a list of individual items each one pointing to one of the X labels. You must get the number of X LABELS correct as the program counts them to see how many lines or groups of items there are to be plotted. If you get it wrong you will be be informed there are Y values missing because it is expecting to have to draw further lines.

TIMELABEL(crd...crd)
These are the tagged items on the X or bottom axis. They are not always time labels. (Plot types 1...6 only)

YVALUE(n,crd...crd)
This function points to the actual Y values to be plotted. The 'n' refers the number of the line to which you are referring. There will be one of these functions for each line.

YMAXIMUM(crd)
The crd must be a value or a reference to a value that specifies the maximum value to be shown on the Y axis. Your choice will be rounded to a suitable nearby value to improve the presentation.

YMINIMUM(crd)
Similarly for a minimum value. Note that if the value you want to round to is the same as value to be plotted you will have to slightly increase the specified minimum value to get the minimum you want.

XVALUE(n,crd...crd)
This function points to the actual X values to be plotted. The 'n' refers the number of the line to which you are referring. There will be one of these functions for each line. (plot types 7...18 only, use TIMELABELS for 1...6)

XMAXIMUM(crd)
The crd may be a value or a reference to a value that specifies the maximum value to be shown on the X axis. Your choice will be rounded to a suitable nearby value to improve the presentation. (plot types 7...18)

XMINIMUM(crd)
Similarly for a minimum value. Note that if the value you want to

round to is the same as value to be plotted you may have to slightly increase the specified minimum value to get the minimum you want. (plot types 7...18)

INPUT/OUTPUT FUNCTIONS

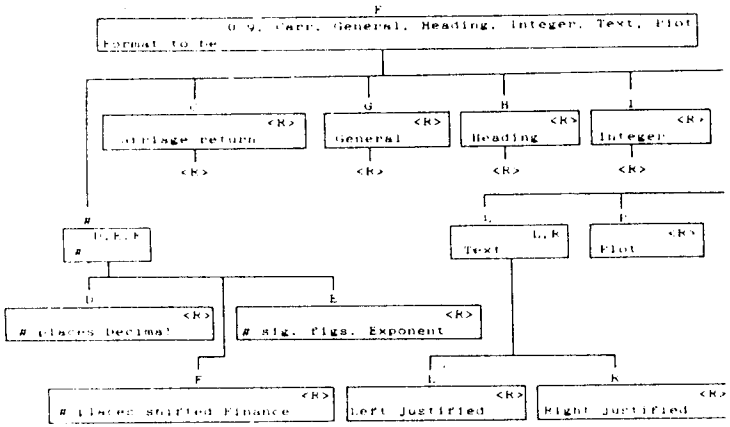
These are two specific functions that allow THE CRACKER to control external peripherals.

IN(port)
Reads an 8 bit port given by the number or cell reference 'port'.

OUT(port,value)
Outputs a 'value' given by a number or cell reference to the 'port' given by a number or cell reference. This should not be confused with the OUT command which sends characters specifically to the printer.

The I/O functions can very profitably be combined with the conditional test IF THEN ELSE, the DO WHILE loop and the DATE and TIME functions for intelligent automatic control of peripheral machinery.

FORMAT COMMAND



The FORMAT COMMAND is used to specify the way in which a cell is to be displayed on the screen. There are three types of FORMAT COMMAND. The first is the GLOBAL FORMAT which sets the format if the entry is started by using either the (or the ' COMMAND. Actually there are two global formats, one for text and one for expressions. On startup the text is set to Text Left Justified and the expression one to General. You can change these formats to a variety of options using the NEW COMMAND.

The second is the DEFAULT COLUMN FORMAT, which specifies the format in which new entries in a column will be displayed if the '.' or > COMMAND is used. Again this format can be changed using the NEW

COMMAND.

The third defines the FORMAT for any new individual cell entry. In this case you specify the format you want just for that cell alone. When the FORMAT for a cell has been defined it will have precedence over the DEFAULT FORMAT.

The format command can has no effect on the way an entry is stored in memory. Once specified the format will stay with an entry even if moved to a new position. This is because it is an entry that has a format and not a cell location. If no format is specified then an entry will automatically pick up the default column format for that column.

Internally the numbers are stored in a binary floating point format. It is not important to understand the details of the internal floating point number merely to know that it will ensure accurate calculations with up to 15 significant figures. Generally you would not want to display all the available accuracy so the program allows you to define just how many decimal places you want to show.

If you do not really know how many decimal places you can ask THE CRACKER for the vague format definition called GENERAL FORMAT. It will display the number in a similar fashion to that of scientific pocket calculators. Thus if the number fits into the space available in its entirety then it will be shown in full. If it is too large or too small then it is converted into the EXPONENT form (see below). This format will make sure that whatever the result of a calculation the result will be displayable.

The maximum number of decimal places is 38. 15 significant figures is the most that can be displayed but internal calculations are done to 16-17 significant figures.

Format types

The format options you have available when entering a value or expression into a cell can each be set to any one of a selection of Format Types detailed below. When specifying a cell format you can choose any of these options, the Global or Default Column formats can be assigned types using the New Format command sequence.

C

CARRIAGE RETURN, this can only be used with the DEFAULT FORMAT. It should only be used with a blank column, usually of width 1 space. Its purpose is to put in intermediate end of line markers in mailing list and database applications. It has no other use.

£F

FINANCE, this outputs all values in balance sheet format layout. The £ represents the number of places shifted you want the results. This is

the same as saying the number of times you want the number dividing by 10 before displaying it. With this facility you can display your answer in thousands or millions. Commas are added every third place in the usual way of presenting financial output. Negative values are enclosed in brackets.

```
                                                    ====A=====
format 0F,  memory  12345.678,   display    12,345.68
   "   0F,     "   -12345.678,      "     (12,345.68)
   "   3F,     "    12345.678,      "          12.3
```

G

GENERAL, this is a general purpose format that will display the value as simply as possible and as nearly in the way that it might be displayed on pencil and paper calculations. Unnecessary zeroes will be removed and if the number is too large or small it will convert to exponent format. The format bears some resemblance to that used on scientific calculators.

```
                                                    ====A=====
format  G,  memory  123.0,           display          123
   "    G,     "   0.123,              "           0.123
   "    G,     "   0.000000123456,     "     0.123456E-06
```

£D

DECIMAL, the £ defines the number of decimal places that will be displayed.

```
                                                    ====A=====
format 5D,  memory  123.456789,  display   123.45679
   "   2D,     "    123.456789,      "         123.46
```

£E

EXPONENT, the £ defines the number of significant figures to be displayed in the decimal part. The exponent form is sometimes called scientific notation. It consists of a decimal number followed by the power of 10 by which it must be multiplied to give the actual number. This power is called the exponent.

```
                                                    ====A=====
format 5E,  memory  123.4567,  display   0.12346E+03
   "   3E,     "    0.001234,      "        0.123E-02
```


I

INTEGER, the value is displayed to the nearest whole number.

```
                                                    ====A=====
      format  I,  memory  567.89,  display          568
```

Tl

TEXT LEFT JUSTIFIED, is for alphanumeric entries. The memory contents will be displayed left justified in the cell.

```
                                                    ====A=====
      format Tl, memory  'TEST',           display        TEST
```

Tr

TEXT RIGHT JUSTIFIED, is for alphanumeric entries. The memory contents will be displayed right justified in the cell. If the cell is smaller than the entry the display will be filled as if it was going to be left justified.

```
                                                    ====A=====
      format Tr, memory  'TEST'              display       TEST
         "   Tr,     "   'THIS IS A TEST',    "     THIS IS A T
```

H

HEADING, is for alphanumeric (text) entries. The memory contents will be displayed in their entirety even if the column is not as wide as the entry. This format can therefore be used for titles when the column width is, at different times, liable to be varied. Adjacent entries will not be displayed if the heading over-rides them.

```
      format H,  memory  'THIS IS A TITLE'

                          ==A== ===B===
                displav  THIS IS A TITLE
```

P

PLOT, will fill up a cell with asterisks to the nearest integer value of the stored number. This is used to provide a very simple histogram representation of data.

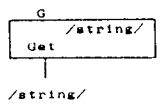
```
                                                    ====A=====
      format P,  memory  5.556,  display  ******
```

Overflow:

If there is insufficient space to display a numerical value then the cell will be filled with asterisks to avoid misleading you with incomplete figures. The same does not occur with text entries (c.f. also headings).

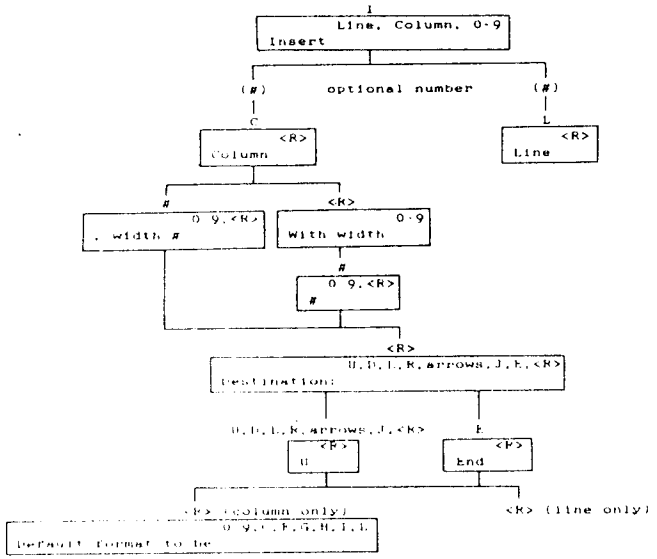
```
                                    ==A==
format 4D, memory  1234.56789,  display  *******
```


GET COMMAND



The GET COMMAND is used to search from the current cell location to the end of the worksheet for a particular string. The string is delineated on the edit line by a single printable character which must not be a letter or numeral. When this delineator or marker is next encountered on the line the string you are typing is assumed to be complete. This system is adopted to give you flexibility over which characters actually are included in the string. The string length may be up to 30 characters not including the delineators. Only the text or expression as it appears on the contents line is searched. You cannot search for displayed values that appear as the result of manipulation by the current formats. The search proceeds across each column and then down a line and so on. If repeated searching is to be used then after initially setting up the string it is merely necessary to put two delineators in, one after the other. The string that was previously used will be re-displayed and form the basis of the following search.

INSERT COMMAND

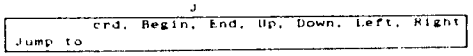


The INSERT COMMAND is used to increase the size of the worksheet. The sheet does not initially exist but you can keep a ready-made blank sheet on file to call in anytime. If you choose to use the Insert command to create your sheet you must start by defining at least one column. With a column inserted you can insert lines. You can insert several columns or lines at a time.

It is also necessary to state exactly where in the sheet you wish the new lines or columns to go. The first lines can only be inserted in front of the cursor (press <R>) or after the cursor (press <E> for END). As the defined sheet builds in size the available options increase.

Increasing the size of the sheet does not use up any of your available memory, this only happens when data is added.

JUMP COMMAND

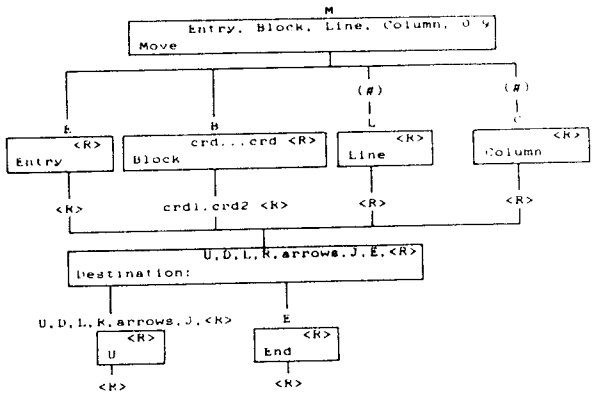


The JUMP TO COMMAND is used to move the cursor directly to the cell you specify or edge of the worksheet. The 'crd' prompt refers to the coordinate cell reference to which you want to go. The other prompts refer to the extremities of the worksheet. So Jump Right means take the cursor across all the columns to the right hand side of the worksheet. This command can be used both as a primary command and in answer to the request for destination.

LOADING A FILE

See the Copy Command

THE MOVE COMMAND

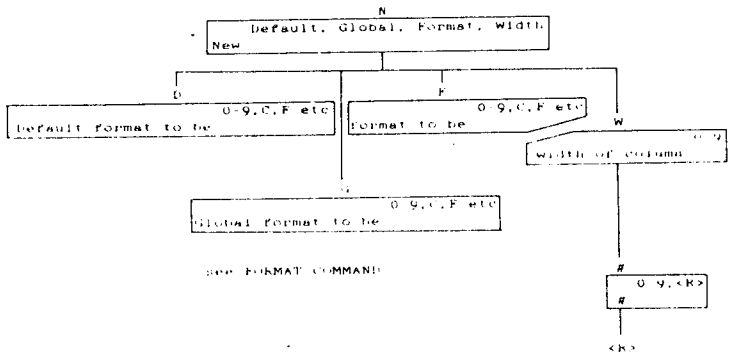


The MOVE command transfers a section of memory to a new destination. Before the movement takes place a check is made to see that the destination area is vacant and if not you will be informed and the command will not be carried out. Because, unlike Copy, the original cell locations are blanked all references to the transferred area, and within the block itself, will be automatically adjusted to the new location.

The same options are available as with the Copy command.

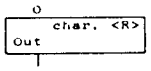
If several lines or columns are being moved an internal check is made that the destination specified will provide sufficient room.

THE NEW COMMAND



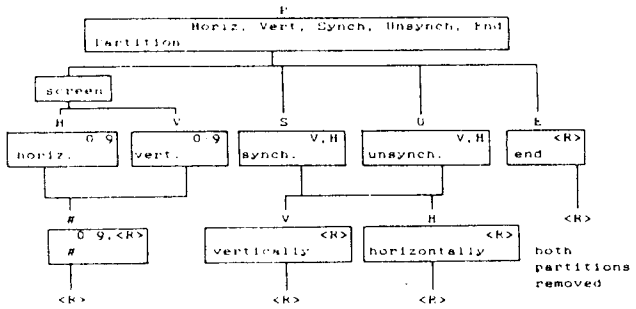
The NEW command defines a change of either width of column, explicit (directly specified) format, default column format or Global format. A new format can only be a change between compatible types for example a change cannot be made between a TEXT and a FINANCE format. This is because one is character information and the other is numerical information. A change in default column format will only affect future entries that do not have their format explicitly defined. No change will be made to existing entries. Please also study the FORMAT command reference.

THE OUT COMMAND



The OUT command is the way you output commands to your printer. You can set up your printer for special features such as condensed print or different character font or a different colour if you have it available. All characters including control codes are valid and so you cannot use the [DEL] key to backspace. If you make a mistake finish off that entry with a RETURN and start it again. Similarly the [ESC] key will not work as it is a code that the printer might validly expect. Finish off each entry with a RETURN. Your characters will be echoed on the screen, control codes all be preceded by a ^. You will find the printer commands in your printer manual.

THE PARTITION COMMAND



The screen may be partitioned either vertically or horizontally to give 1,2, or 4 available windows and the movements of the cursor locations within these windows may be defined as synchronized or unsynchronized for both the horizontal and vertical partition. Synchronizing means that movements in one window will be matched by equivalent movements in the other. Without it the inactive window will be unchanging.

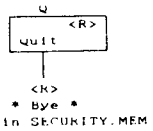
The horizontal partition location can be specified by taking a number from the grid that will be displayed on the screen. The vertical partition is similarly defined. Partitions may be removed by the PE partition end command sequence.

To jump the cursor from one window to another the slash commands are used. / will jump left to right and vice versa and the \ will jump up and down between windows.

PRINTING A FILE

See the Copy Command

THE QUIT COMMAND

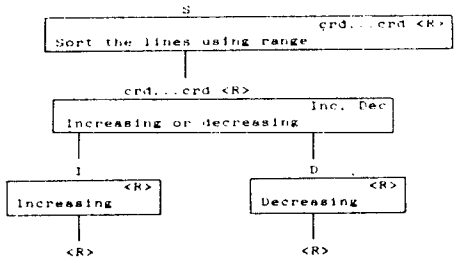


The QUIT command clears the screen and makes a return to your DOS. A copy of your worksheet is always saved on file SECURITY.MEM. Experience shows that without this safety precaution it would not be long before you lose valuable work by mistake. Remember to rename this file at the earliest opportunity to avoid accidental overwriting.

SAVING A FILE

See the Copy Command

THE SORT COMMAND

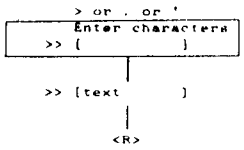


The SORT command allows sorting of lines using any specified part of any particular column as the key. The sort may be by increasing or decreasing values and may be carried out on both numerical values and characters. Every column is sorted with respect to the range to keep the integrity of the lines with the important eception that formulae references are not moved or adjusted. Normally this can be a very useful effect but you should avoid sorting where you have mixed formulae and constants in any column over the range or you may get very confusing results.

Note that mixing of text and numerical values in the sort should be avoided as this will give indeterminate results. Capital letters and lower case letters are taken to be of equal value. With text sorts numbers come after letters followed by blanks, punctuation marks and finally control codes. All punctuation is taken to be of equal value. Remember that any numbers that are entered as text characters can sort in unpredictable ways, for example 8 will sort as a higher value than 24.

If text or blank entries are within the specified numeric range, or vice versa, they will tend to sort to one or other end of the range.

TEXT ENTRY



If you use the '.' COMMAND with the default format set at Text, or use the FORMAT COMMAND set for Text, or use the Global Format form of entry with the ' COMMAND, then you will be set up for entering text. A marker will be put on the edit line to show the current column width, but you are free to over-write this with a longer entry.

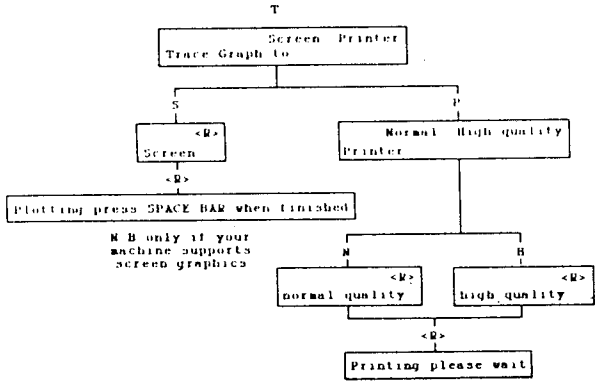
The prompt ENTER CHARACTERS will come up. Any number or character may be inserted and if a mistake is made the [DEL] key (or maybe the backspace on your machine) may be used to backspace. If the entry has been completed and you want to move to an adjacent cell and insert information there, then you can use the arrow keys instead of the RETURN key.

Pressing an arrow key once will move the cursor into the relevant adjacent cell and the program will be set up for entries at that location without any further key pressing. The format will remain the same as for the previous entry. The one key movement therefore bypasses the RETURN, cursor movement and the > key for ENTRY mode. It is for this reason that the backspace key cannot be used while entering text or expressions because the backspace key and left pointing arrow often give out the same command to the computer producing ambiguous results.

On completion of text entry it is inserted into the cell location. The memory remaining is changed on the screen.

TRACE GRAPH COMMAND

<flow chart on the new handout



The Graphics features of THE CRACKER work by using a set of special functions to point to cells containing the data to be plotted and the text for the labels.

Note that where 'crd' is specified in these functions you must enter a reference to a cell coordinate. You can't use numerical or expression equivalents except where stated.

Here are the functions:

TYPEPLOT(crd)
The type of plot you want. Give a value between 1 and 18 in a cell reference to this value. The plot types are detailed later.

MAINTITLE(crd)
The cell coordinate is a pointer to where the main title is to be found.

SUBTITLE(crd)
The same idea for the subtitle.

YTITLE(crd)
The pointer to the title up the Y axis on the left hand side.

XTITLE(crd)
Where the X axis title is. This is the one at the bottom.

XLABEL(crd...crd), or xlabel(crd,crd,crd,etc.)
This points to the X labels which are the legend box items on the right.

Unlike the other functions in this list the range can also be a list of individual items each one pointing to one of the X labels.

The xlabels function also implicitly signals to THE CRACKER how many plot lines or plot types you wish to include.

TIMELABEL(crd...crd)
These are the tagged items on the X or bottom axis. They are not always time labels as such but must be legends of some kind rather than actual values. (Plot types 1...6 only)

YVALUE(n,crd...crd)
This function points to the actual Y values to be plotted. The 'n' refers the number of the line to which you are referring. There will be one of these functions for each line.

YMAXIMUM(crd)
The crd must be a value or a reference to a value that specifies the maximum value to be shown on the Y axis. Your choice will be rounded to a suitable nearby value to improve the presentation.

YMINIMUM(crd)
Similarly for a minimum value. Note that THE CRACKER itself rounds the minimum and maximum values to those that it feels is best suited for the particular graph. You mat have to experiment with this value to get the minimum you require.

XVALUE(n,crd...crd)
This function points to the actual X values to be plotted. The 'n' refers the number of the line to which you are referring. There will be one of these functions for each line. (plot types 7...18 only, use TIMELABELS for 1...6)

XMAXIMUM(crd)
The crd may be a value or a reference to a value that specifies the maximum value to be shown on the X axis. Your choice will be rounded to a suitable nearby value to improve the presentation. (plot types 7...18)

XMINIMUM(crd)
Similarly for a minimum value. Note that THE CRACKER itself rounds the minimum and maximum values to those that it feels is best suited for the particular graph. You mat have to experiment with this value to get the minimum you require.
(plot types 7...18)

PLOT TYPES

Business (time labels rather than XVALUES):
1. Bar chart (histogram)
2. Stacked bar chart
3. line chart
4. Area chart
5. Hi-lo chart
6. Pie chart

Statistical, engineering and scientific:
7. X:Y line joining points
8. LogX:Y line joining points
9. X:LogY line joining points
10. LogX:LogY line joining points

11. X:Y points only
12. LogX:Y points only
13. X:LogY points only
10. LogX:LogY points only

15. X:Y points and best fit line
16. LogX:Y points and best fit line
17. X:LogY points and best fit line
18. LogX:LogY points and best fit line

The best fit lines of plot types 15...18 are obtained by the least squares method.

Remember that the above are FUNCTIONS with arguments and as such must be entered into cells that have a numerical format EVEN IF they refer to some textual data such as a label. The cell that contains the label itself must of course be set to a text format.

If any errors are encountered while using the graphics an error message will be given and a return will be made to the spreadsheet.

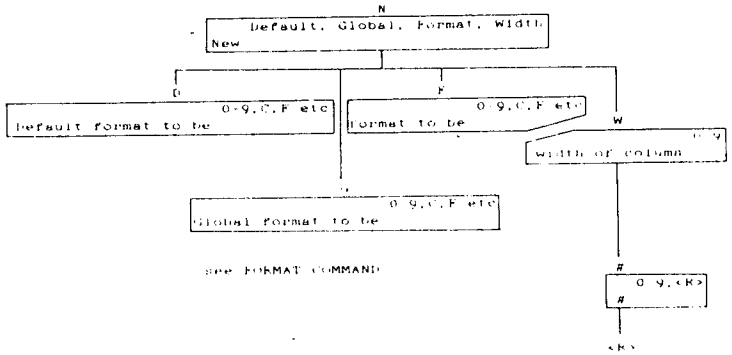
Z80 machines only

This is an early version of the graphics program and so if you find you can not return to the main program don't worry you work to date will be found on the file SECURITY.MEM (16 bits versions are not overlaid)

The GRAPHICS utility is loaded by THE CRACKER and when it has finished it re-loads THE CRACKER. To make sure you end up at the point you started THE CRACKER must be on the original disk unit and user area. You don't have to do anything but if you have changed disk units or user areas don't be surprised if you have been changed back again. When you call TRACE you must therefore have the original disk with THE CRACKER on it in the original drive.

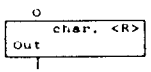
The graph drawing functions make considerable use of disk overlay files and also make temporary security copies of the data. You may find that you have to make some adjustments to the layout of files on your discs, particularly on single drive systems.

THE NEW COMMAND



The NEW command defines a change of either width of column, explicit (directly specified) format, default column format or Global format. A new format can only be a change between compatible types for example a change cannot be made between a TEXT and a FINANCE format. This is because one is character information and the other is numerical information. A change in default column format will only affect future entries that do not have their format explicitly defined. No change will be made to existing entries. Please also study the FORMAT command reference.

THE OUT COMMAND



The OUT command is the way you output commands to your printer. You can set up your printer for special features such as condensed print or different character font or a different colour if you have it available. All characters including control codes are valid and so you cannot use the [DEL] key to backspace. If you make a mistake finish off that entry with a RETURN and start it again. Similarly the [ESC] key will not work as it is a code that the printer might validly expect. Finish off each entry with a RETURN. Your characters will be echoed on the screen, control codes all be preceded by a ^. You will find the printer commands in your printer manual.

SOME STATISTICS

max length	MAINTITLE	24	chars
	SUBTITLE	35	
	YTITLE	25	
	XTITLE	25	
	TIMELABEL	16	
	XLABEL	24	
no of	TIMELABELs	32	
	XLABELs	9	(no of lines) possible on one chart
	YVALUEs	9	
	XVALUEs	9	

If you use lengths longer than specified above they will be truncated or ignored.

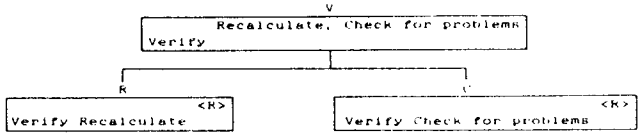
PLEASE NOTE

If you do any EDITING of the plotting instructions or make any changes that do not force a recalculation, you may end up with an error message or values that does not seem to be true. This is most likely if you are using direct values rather than references to cell coordinates.

The surest way to handle plotting functions that do not contain cell references is to overwrite them rather than edit them.

The instructions for plotting are worked out whenever a recalculation is done so if no recalculation has occurred the latest instructions may not have been properly determined. If you get such an error message just use the ! FORCE RECALCULATION COMMAND and try the plot again.

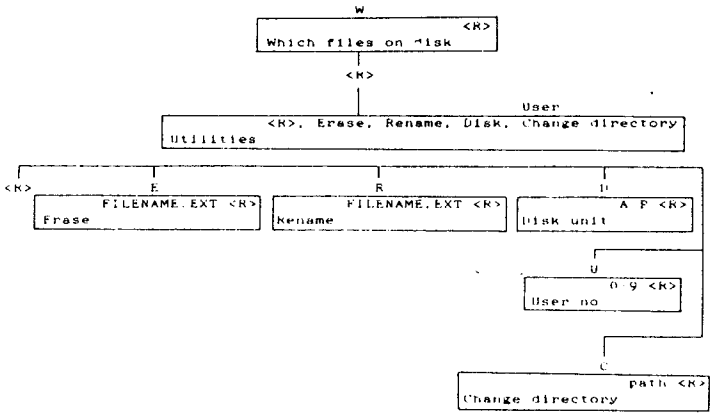
THE VERIFY COMMAND



The VERIFY AND RECALCULATE command option is used to cause a full recalculation including the constants within your worksheet. That is in contrast to the ! command which only recalculates expressions. Normally force a recalculation with the ! command.

If you ask for VERIFY AND CHECK ON PROBLEMS then the structure of the worksheet will be analysed to look for inconsistencies and if any are found an attempt will be made to fix them. This is a safety precaution to ensure as much as possible that you are never left with an unworkable spreadsheet.

THE WHICH FILES COMMAND

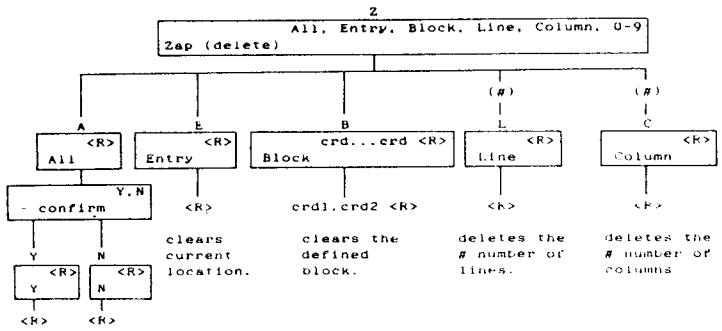


The WHICH FILES command is used to look at files associated with THE CRACKER on your disk and also to give the basic file handling operations you may want. They include erasing files, renaming files, changing disk units and moving around the separate parts of the discs.

The DOS file directory will be read and all the compatible files, i.e. the screen. After this you will be able to do any of the file management operations you want to. Obviously there are slight differences depending on which DOS you use. For MSDOS you have the ability to move THE CRACKER freely between sub-directories. These directories will be given at the start of the file list together with the name of your current directory. For CP/M the current user number will be displayed at the head of the file list.

Remember if your system supports passwords then you will have to give your allocated password in order to erase or rename any of the files.

THE ZAP (delete) COMMAND



ZAP deletes the defined entries from memory. If the command refers to an entry or a block then the command acts just like the BLANK command. For references to Line, Column, or All then the size of the worksheet is actually reduced by the command. Deleted entries CANNOT be retrieved so if in doubt first copy the memory to a file before embarking on complex rearrangements.

Before the deleting is carried out the program will check whether any of the items to be deleted are referred to elsewhere in the worksheet. If cross-references are found the command will not be carried out. You will be allowed to use Blank in these circumstances because the cells in question remain on the sheet and can be filled with a row of '?' to signal that data is missing.

THE ! (force recalculation) COMMAND

The ! FORCE RECALCULATION command has two main purposes. The first is to cause a calculation when the AUTOMATIC CALCULATION is off. It is a minimum recalculation and therefore takes less time than the VERIFY AND RECALCULATE command.

The second usage is for solving iterative problems such as simultaneous equations or other circular reference problems. In these types of problems there may be two cells which refer to each other. Initially they cannot both be valid but on repeated recalculation they will approach true answers.

A typical example could be the equation X=COS(X). X is on both sides of the equations and cannot be extracted, but by deliberately entering the equation into the worksheet as a circular reference the answer can be got after just a few format recalculations. To do this example put COSR(A1) in cell B1 and B1 in cell A1. After about 20 recalculations it is starting to settle down. It is relatively simple to automate this type of calculation with a DO WHILE looping function.

THE * MACRO COMMAND

The * MACRO command is initiated by typing an asterisk, *, followed by a reference to the coordinate of the cell in which the MACRO is stored.

The purpose of this command is to call in a predefined set of commands. Just type the appropriate key press sequences for your desired commands into any text format cell.Where you would want to put a RETURN use the @ symbol. It is valid to finish a sequence with another MACRO command or even a reference to the same macro. With a self-repeating reference you can create macros that go on repeating until an error is found.

This command is ideal for repetitive changes of format down a whole column for example. The macro NFG@D*A1 placed in cell A1 will change a format to GENERAL and then move the cursor down. The *A1 at the end means that it is called again and so the cell underneath it is changed and so on. This will stop when an attempt is made to change a cell outside the worksheet.

The Macro can be used to create a loop of direct commands, which differs from a loop of functions such as you may use in an expression.

MORE DETAIL ON ERROR MESSAGES

A-Z, or 0-9

Only a letter or numeral may be used at that point.

Ambiguity (see manual pp xxxx)

The order of calculation number can not be resolved without ambiguity. The expression must be preceded by a reference to the cell that created the SET( ) to which this cell refers. See index under 'ambiguities'.

Argument too large

If calculated the EXP( ) of this value would be greater than 1E+38.

Argument > 1.0

ASIN, ACOS, ASINR, ACOSR can only have arguments between 0 and 1.

Cannot multiple move there

There is not sufficient space at your specified destination to move this group of lines or columns.

Column first

Please insert a first column before you try to enter lines.

Comma only

You seem to have confused the layout of this function's argument.

Def function badly set up

The layout of the defined function is wrong.

Disk directory full

You have more than the allowable number of files on the disk already. THE CRACKER cannot therefore write to the disk. Use the WHICH FILES command to erase some of the files that you don't need. Start with those that have a .BAC extension.

Disk space full

There is no more room on the disk. If you are trying to save your work put in a blank disk and start again. The disk must be blank if you are using CP/M 1.4 or 2.2 or an early version of CDOS as the work on the disk may be indiscriminately overwritten.

Division by zero

You are trying to divide by zero which would give infinity. THE CRACKER for all its power cannot handle that. This message may appear if you have blanked some cell entries.

Do split first

You have used one of the slash commands before the partition has been made on which these commands work.

DO( ) needed with WHILE( )

You cannot use the WHILE( ) function on its own. Part of the job it does is to seek out the DO( ) on the same line.

E+38 maximum

The maximum exponent that is available in the floating point numbers used in THE CRACKER is +38.

Entry too long

The maximum entry width is the free area in the middle of the entry line. This varies with the make of computer you are using. The error message appears if you try to type beyond this limit. The absolute maximum a cell can hold is 127 characters.

ERROR called from <crd>

This is an error you yourself have called by using the ERROR( ) function.

FILENAME.EXT not on this disk

The file named cannot be found on the disk. Use the W for Which files command to find which files are available.

File, Print or Mail Label

You asked for an option other than one of these after Copy All.

Function (((((5))))) max

You may only nest functions and brackets to 5 levels.

Logical expression only

The IF( ) function must resolve to a value of TRUE, FALSE, -1 or 0.

Missing "("

You have too many right hand brackets.

Missing ")"

You have too many left hand brackets.

Move cursor

The command you are trying to do does not make sense unless you actually move the cursor.

Must be a constant

You cannot overwrite a formula with a SET( ) or INIT( ) function.

Must be a letter

A column letter is expected.

Must be "logical" (-1 or 0)

The required argument at this point must resolve to a logical value.

Must be one of +-*/ <>=%[]|), or <R>

One of these operators are required.

Must not refer to this cell

The cell that contains the DO( ) function itself is in the range of its own argument. This is a circular call that cannot be resolved.

Negative argument

Cannot have a negative argument to a logarithm or square root.

No function "FUNC("

This function you have entered is not one of the built in functions.

No such column

You have made a reference to a column outside the worksheet area currently defined.

No such line

You have referred to a line outside the worksheet area.

Not enough memory

There is not enough memory left to read in this file.

Not enough space

Part of your destination area would be outside the worksheet as currently defined.

Not found in this range

The LOOKUP( ) or INTERP( ) first argument was not present or enclosed by any two values in the list that you have specified.

Not in marked zone

You are trying to partition outside the area marked by a grid on the screen. Reset using Delete and re-position the screen window if required.

Number or value missing

With LOOKUP( ), CHOOSE( ) and INTERP( ) none of the items in the list can be blank.

Odd number of values required

Simpson's rule works on an odd number of values only.

Operator at end

You cannot end an expression with an operator.

Out of memory

The memory available for entering data is all gone. Your work to date will not be damaged. Try to split your work and write an independent portion to a file so that you can later consolidate the results.

Reference outside worksheet

This cell reference is outside the area of the worksheet as you have currently defined it.

Result more than 1E+38

You have done a mathematical operation that gives too large a result. This is probably because of an error in one of your constants.

Second "."

You can only have one decimal point in a number.

Separate these operators

You have two operators together. In addition if you have two brackets together i.e.)(then you must separate them with a *, thus)*(.

Should be one of +-*/ <>=%[]|), or <R>

These are the valid operators at this point.

Stack overflow, shorten entry

There are too many pending operations. Cure this by splitting the expression into two or more cells.

Text reference in an expression

In this expression you have made a cell reference to a text entry.

Text/Value change

You are trying to change a formula or constant into a text entry or vice versa.

The range must increase

Only ranges that extend over at least two columns or lines are really ranges.

Too long

The String used by the SEARCH command can only be 30 characters long at maximum.

Too many chars

Your filename has more than 8 characters.

Worksheet too small.

You will have to expand the worksheet area before you can copy any data in from a file.

Wrong conditional layout

You have confused the layout of this conditional entry.

Wrong range layout

Zero argument

You cannot have a logarithm of 0.

0...9 or "."

A numeral is expected at this point (the start of an exponent)

15 sig figs max

THE CRACKER calculates to 16-17 significant figures but for security always rounds down to 15.

255 lines max

The maximum number of lines in a worksheet is 255.

38 places max

The floating point number system of the THE CRACKER only goes down to E-38.

52 columns max

The maximum number of columns is 52, they are marked A-Z then a-z.

.TXT, .DIF, .DAT or .MEM

Only these file extensions are valid if you are trying to read in a file.

"end" wrong here

It does not make sense to have "end" as the destination here.

"(" or "," first

A range must always be preceded by a bracket or comma. It cannot be used as part of an expression.

"£" cannot follow a range

You cannot get the current value of a range.

"?" is wrong after a coordinate

This character is not a correct following operator.

"?" is wrong here

This character is not one of the options you were offered in the prompt list.

"?" should be a numeral

Only a numeral would be valid here.

<B22> is used in <C3>

You have attempted to remove or overwrite an expression that is referred to in another expression. To do this would stop THE CRACKER being able to do a full calculation.

%% 1 Entry destroyed in calculation
%% 2 Column details corrupted
%% 3 Internal coded format error
%% 4 Entry not found to format
%% 5 Entry not found for calc number
%% 6 Index table corrupted
%% 7 Main storage corrupted
%% 8 Cannot delete index table item
%% 9 Negative number of index entries
%% 10 Zero length entry
%% 11 Does not exist
%% 12 Command corrupted
%% 13 Memory corrupted, saved, check copy carefully
%% 14 Opening space for index table error
%% 15 Invalid binary number format
%% 16 Number formatting error
%% 17 Number formatting error
%% 18 Number stack corrupted
%% 19 Operator decoding error
%% 20 System error
%% 21 Look-ahead buffer corrupted
%% 22 Disk write record error
%% 23 System error

These are system error flags, that signal that a serious error has occurred in the working of the program rather than in the entry that you have typed. All being well you should never see one however they are the long stops in the event of system bugs. If you get one please note the exact circumstances of how it occurred and inform your supplier. As a general rule your work will not be damaged even if you find such a bug. THE CRACKER will automatically check that your work has not been damaged. If it has then it will be filed automatically. Just press the [ESC] key in the usual way and you will be able to carry on.

APPENDIX

USING THE CRACKER ON YOUR AMSTRAD SYSTEM

It is important to realize that just because the price of computers has been reduced it does not mean they are easier to use. You have enormous power at your fingertips but you are going to have to work your way into things. Firstly you must get used to CP/M which is the program that has overall control of your machine, secondly THE CRACKER.

What seems unfamiliar complicated and frightening now will seem trivial in a few days as you get used to things. If you can follow these few instructions to get you running THE CRACKER you will be over the first hurdle. Once THE CRACKER is going it will protect from almost everything that you can do wrong. Hold your nose, don't worry what it all means at the moment, and jump in!

WHAT IS CP/M?

THE CRACKER runs under CP/M+ on the AMSTRAD computers. CP/M is known as a computer operating system. Every computer needs an operating system - it is the Operating System that figures out what you are trying to say when you type something at the keyboard, keeps track of which disk drive you are using, sends information to the screen etc.

On the CPC6128, CP/M is supplied as an alternative operating system to that which normally takes charge when you first turn on your computer. CP/M is a standard system which allows programs to be made available on a range of different computers with minimal additional conversion or installation work.

The PCW machines have no inbuilt operating system and are effectively unable to do anything until CP/M is loaded.

You are given some information on how to use CP/M in the manual supplied with your computer. You are advised to find time to familiarize yourself with the basic details of using CP/M before rushing into THE CRACKER. The minimum necessary knowledge can be picked up in a very short time. If you are likely to be spending a considerable amount of time using CP/M software you would be advised to look for a more comprehensive book on the subject and perhaps a second disk drive.

CARE OF DISKS

The 3" CF2 discs used with Amstrad machines are very much more robust than most typical computer discs. Even so we must not be tempted into treating them carelessly.

Keep them away from any strong magnetic fields:- e.g. from normal unshielded televisions, X-Ray systems - and more mundance hazards such as coffee cups, buckets of water etc.

NEVER be tempted to pull back the metal protective flap to touch the actual magnetic surface itself.

Consult the section on disks in your computer operating manual. Locate the tag known as the Write Protect tab and set this on any discs that you know you definitely will want to preserve the data - this prevents accidental overwriting under any circumstances.

Investigate the use of the SET utility explained in your CP/M manual for making selected files READ ONLY.

Most importantly *ALWAYS* keep at least two updated copies of any important data. Accidents do happen and this is the only reliable safeguard.

CPC 6128
The program disk you are supplied with will consist of some demo files and the program files CRACKER.COM, CRACKER.OVR and CRACKER.HLP.

In order to use a data disk with a single drive CPC6128 machine you will have to first format a disk with the DISCKIT3 utility and then use FILECOPY or PIP to take a copy of CRACKER.OVR from THE CRACKER master disk and put it on your data disk. If you want online help as well you will also have to copy CRACKER.HLP onto your data disk. This is not essential if you want more workspace.

To get going put the disk with CRACKER.COM on it into the drive and type CRACKER.....

A>CRACKER

You can now take out the disk and put in your data disk with CRACKER.OVR on it. If you want to use graphics you will have to have both CRACKER.COM and CRACKER.OVR together on your data disk.

Having all three files, OVR, COM and HLP on one disk will leave very little room for your working files so you are advised to do without the help facility when using graphics. Read the manual instead.

An alternative approach which is slightly more awkward but that makes more efficient use of your data files is to use the 'virtual drive' capacity of the CP/M supplied. In this way you can load THE CRACKER on drive A, but change the logged (current) drive to drive B. Although you only have one disk drive the computer will allow you to treat one of your discs as if it runs on drive A and one as if it runs on a second (imaginary) drive B. You will be prompted whenever it is necessary for you to change discs in the drive.

Because THE CRACKER keeps much of the program and all of the data in memory at once you will find that although you will have to change the discs a few times when loading once you have done so there will be less need for disk changing than with many programs.

However if you using the on-line help facility or the Trace graph option, you will probably find that there is so much disk swapping involved that the disadvantages outweigh the value of the space saved.

In practice you may also find that several attempts are required to get the program to recognise when the discs have been changed. No harm will result but the error messages that appear can be confusing and misleading.

Please don't use the write protect facility on the disks. If you try to write to a write protected disk it won't be possible and you may get a random error message such as 'disk full' or 'file not there'.

Printing on the CPC 6128

It is possible to send control codes to your printer to select certain special effects (provided that you know what they are for your particular machine) in order to select certain features such as condensed or high quality print if available. THE CRACKER command for doing this is the OUT command.

However when attempting to produce a screen copy from the Trace Graph option you will find that the results may not be correct unless you have an 'Epson compatible' printer.

PCW MACHINES ONLY

KEYS:

Amstrad have renamed a couple of (otherwise standard) keys on the PCW machines. The control [CTRL] key is now called [ALT] and the escape [ESC] key is now called [EXIT]. Note this when you read any of the manuals. Some of the keys you will require are apparently not available. These 'missing characters can in fact be obtained by using the extra key together with a normal key. The particular combinations are to be found in the section I.2 page 105 of the CP/M operating system section. The best way to start to use THE CRACKER is with the [CAPS LOCK] on. This is got by pushing the [ALT] and [ENTER] keys simultaneously. Press them a second time stop the [CAPS LOCK]. You may not need to turn this feature off unless you are entering text data or are expanding the size of the sheet beyond 26 columns.

GETTING GOING.

The first step is to create a working copy of your program disk.

While the machine is OFF, put in the distribution CP/M PLUS disk with 2 on the top left hand corner. Switch on the machine and you will be in CP/M. Note do not start up with the LOCO SCRIPT disk.

Take two new disks, one for a copy of your supplied disk and the other for a blank work area (data) disk. Start by formatting (preparing for use) a data disk. Type DISCKIT followed by [RETURN]. Follow the instructions for 'format'. When this is done press the key to 'copy'. Follow the instructions again and this time put in THE CRACKER master disk when you are asked for the disk to copy from. Change disks when asked to. When finished put THE CRACKER master disk in a safe place and don't use it. It is for emergencies only.

The best way to run THE CRACKER is to use PIP (copy) to transfer the file called CRACKER.OVR and CRACKER.HLP to the M: disk, this is the memory disk which is very fast and gives you more space to work with on your floppy disks.

To do this put in your CP/M PLUS disk (marked 2) and type PIP followed by [RETURN], you will see

```
A>PIP
CP/M 3 PIP VERSION 3.0
*
```

Now take out your CP/M PLUS disk and put in THE CRACKER disk. Type

```
M:=A:*.OVR
```

followed by a return. You will see

```
*M:=A:*.OVR

COPYING -
CRACKER.OVR
*
```

If you want built in help messages you will also need to transfer the HELP file. However don't do this if you want use the graphics as you will not have enough M: disk space. It doesn't matter if you leave it out. Don't use the graphics until you are familiar with THE CRACKER and CP/M PLUS.

Now type M:=A:*.HLP, like this

```
*M:=A:*.HLP

COPYING -
CRACKER.HLP
*
```

Finish off by pressing the [STOP] key.

You now have a copy of the overlay and help files on the memory disk which means you will have no delays when they are used.

Put in your CRACKER disk and type CRACKER

```
A>CRACKER
```

When this has got going on the screen you can take out THE CRACKER disk and put in your blank (work area) disk. This means you will have plenty of space to write your work out to.

MEMORY DISK

For those who are unclear a memory disk, known as drive M on your system, is a special area of memory that is made to behave exactly as if it was a disk drive. It can hold several program files at once, access each of these selectively, respond to normal disk filing commands and so on.

It has the advantage that it responds very many times faster than an ordinary disk drive but there is also the disadvantage that when the machine is turned off all information held in the M disk is forgotten.

Many CP/M programs work best with two disk drives, one for the program files and one for the data disk. Because of its volatile nature it is a sound general rule to use the memory drive for running program files and the real disk for storing data. In that way none of your work should be lost in case of an accidental power loss.

The Amstrad memory discs come in two sizes. The unexpanded form is 112K - which is unfortunately too small to hold quite the same amount of information as a real disk. This can mean that we have to be very selective about which program files we can put into the memory.

If you have the EXPANDED M: disk don't worry about this just copy all THE CRACKER files onto to the M: disk by typing PIP M:=A;*.*. Your data files can be loaded from and saved to a blank data disk in drive A by either changing the logged drive using THE CRACKER's Which Files command or by specifying drive A when entering a filename e.g. A:MYWORK. You could use drive M for the storage of your workfiles but don't forget to copy any files you want to save back onto a floppy disk at frequent intervals or you will lose them when the machine is switched off or there is an accident.

If your memory disk has a size of of 112k then it has not been expanded and you should follow the instructions in the next section.

To speed things up it is recommended you put at least a copy of the overlay file CRACKER.OVR onto the M: disk in user 0. THE CRACKER will look for it there first. This also allows you to take out the original disk and put in a data disk. But not if you want to use TRACE. If necessary you must put the original disk back in temporarily as described above. If you get it wrong don't worry you will be told what's happened and you will always be able to recover the situation.

AUTOMATING THE CRACKER LOADING

If you study your computer manual under the sections that deal with the SUBMIT file and CUSTOMISING YOUR SYSTEM you will be given help on how to produce programs that save you from a lot of repetitive typing.

In brief here is a method that can be used for loading THE CRACKER.

Firstly put your CP/M disk into the drive and type this

```
A>PIP
```

you will see the prompt

```
CP/M 3 PIP VERSION 3.0
*
```

Now type

```
B:=A:PIP.COM
```

When prompted to insert the disk for drive B put your working copy of the program disk in the drive. Press a key.

Next type

```
B:=A:SUBMIT.COM
```

and follow the prompts.

You will now have put a copy of the two files, PIP.COM and SUBMIT.COM onto your work disk. PIP is a program that lets you send files between different discs (as well as the screen, printer etc.). SUBMIT is a program that will perform a range of commands saved in a file as if they had been typed in at the keyboard.

The next step is to use a text editor to create a file on your program work disk called CRACKER.SUB. Locoscript can be used if you have version 1.2 or later (see the option for creating plain ASCII files), Newword can be used in Non-Document mode or you can use the supplied utility ED.COM as a last resort. See your manual for details.

Make the file contain these lines

```
PIP M:=A:CRACKER.OVR
PIP M:=A:CRACKER.HLP
CRACKER
```

All you then have to do to load THE CRACKER from your work disk is to type CRACKER.SUB and the above stored commands will be executed in turn.

SPECIAL INSTRUCTIONS FOR PRINTED GRAPHS

Unless you have the expanded memory disk don't be tempted to put CRACKER.COM and CRACKER.OVR into the memory disk for plotting. The only successful way to do printed graphs is to have a copy of CRACKER.COM on your data disk and CRACKER.OVR on the memory disk. Keep THE CRACKER on one of the floppy disk drives. The reason is simply that THE CRACKER writes temporary files while printing plots and and there is not enough space on the memory disk.

The GRAPHICS utility is loaded by THE CRACKER and when it has finished it re-loads THE CRACKER. To make sure you end up at the point you started the THE CRACKER must be on the original disk unit and user area. You don't have to do anything but if you have changed disk units or user areas don't be surprised if you have been changed back again. When you call TRACE you must therefore have the original disk with THE CRACKER on it in the original drive.

SPECIAL PRINTER EFFECTS

The extra large screen supplied with the PCW series of computers is very useful for displaying the data in your sheet but it can have one or two unexpected side effects.

Most notably the CTRL-P option which is used for printing a snapshot of the screen will produce unintended results because the printer is only able to print 80 columns of normal size characters on one line. The overlapping 10 characters will be forced onto a second line.

If you wish to produce a printed screen snapshot you must first select condensed print by sending the appropriate control commands to the printer (see below). Alternatively the screen image can be printed using the EXTRA and PTR keys in conjunction.

If you wish to make copies of the screen using these keys you may wish to investigate the option of having inverse screen colours that can be selected using the INSTALL file supplied on your program disk. The INSTALL files should be present on the same disk as the CRACKER.COM file - once run by typing A>INSTALL at the CP/M prompt the program will make direct changes to THE CRACKER.COM file in response to the choices you make.

You can have condensed or expanded print when printing out your work. You can either give the commands to the printer directly using the out command or your can build the commands into text entries.

Using the OUT COMMAND on the PCW8256

The following key combinations send the appropriate codes to the printer once the Out command has been selected.

Condensed print, type: [ALT]-O then [RETURN]
(that is you hold down the [ALT] key and type O, must be capital)
Normal pica print, type: [ALT]-R then [RETURN]

Enlarged text, type: [ALT]-N then [RETURN]
Cancel enlarge, type: [ALT]-T then [RETURN]

These four [ALT] keystrokes may also be put in text entries. You can't EDIT them in, but you can enter them as you would a normal text entry. They will be be displayed as if they are blank spaces but the space won't appear on the printer.

There are other printer commands you can give and these are on pages 131 and 132 of the 8256 manual. Where they say [ESC] you just type the [EXIT] key. For example, you can get Elite text using the OUT COMMAND by pressing the [EXIT] key followed by a capital [M] and finishing off the command with a [RETURN]. You can't backspace to correct using the OUT COMMAND. If you make a mistake press [RETURN] and do the operation again.