

## CONTENTS

Introduction	3
Fundamentals	5
Cursor controls GO TO command	
Entering data (the Dimensions model)	8
EXPRESSION command CALCULATE command Blocks and ranges LABEL command	
Entering text (the Bank Account model)	13
WIDTH command BLANK command NEW command REPLICATE command	
More advanced commands (the Journey Planner model)	18
QUERY command SUM command FORMAT and UNIVERSAL FORMAT commands	
Business planning (the Cash Forecasts model)	24
Forecasting cash flows Percentages and factors	
Some commands for calculating, printing, and saving	30
CALCULATE ACROSS command ORDER command TYPE command PRINT command SAVE command	
How to deal with a crisis	34
CONTINUE GO TO go and GO TO rs Overflows Expression length Advanced saving and verifying	
Final reminders and suggestions	40
Postscript (the spreadsheet story)	44
SUMMARY OF COMMANDS	46

Copyright (c) 1983 by MiCROL. All rights reserved. See cassette copyright notice for further information.

Published and printed in England.

MiCROL (UK)  
38 Burleigh Street  
Cambridge

Subject to statutory limitation MiCROL accepts no liability for loss, loss of profit, or consequential damage arising from the use of THE SPREADSHEET or from information contained in this manual. It is the purchaser's responsibility to ensure suitability for purpose.

## Chapter 1 Introduction

### What is a Spreadsheet?

A spreadsheet is basically a table of figures, complete with headings and footnotes, which is used to:

- \* summarise a mass of information (such as the sales data for a department store), or
- \* set out complex calculations (such as costings or estimates) in an easy-to-follow form.

Spreadsheets can be prepared by hand, preferably on the special stationery known as analysis paper. They can also be created inside a computer's memory, using programs like the MiCROL spreadsheet program for this purpose.

Spreadsheet programs are one of the most popular software products in microcomputing. They are widely used in business and industry, especially in engineering and financial planning.

They also have many personal applications. You can use them for checking up on your bank account, for planning your holiday, for estimating how much paint you need to decorate a house, and many other tasks. And once you have constructed a spreadsheet for a particular task, you can store it on tape (or disk) for reuse next time you face a similar problem.

As a result, an understanding of spreadsheet programs is fast becoming essential for anyone who uses a microcomputer at work, at college, or at home.

The MiCROL spreadsheet program is ideal for beginners and experts alike, since it is clear and easy to use, and remarkably sophisticated in view of its modest price.

### The MiCROL Spreadsheet Program

The MiCROL spreadsheet program is designed specifically for small personal computers.

It includes many facilities which are found in spreadsheet programs written for larger machines. It also has some special features of its own, which help it to cope with the smaller screens and lower operating speeds of a home computer.

The program supplied on the accompanying tape contains four

ready-to-use spreadsheets, of differing levels of complexity, which illustrate many of the techniques explained in this manual. Unless you are already familiar with spreadsheet programs, it is best to have the program running when you first go through the manual, so that you can try out each feature as you read about it.

### Loading the program and models

The main spreadsheet program (with the built-in demonstration spreadsheet) is recorded on side A of your cassette. To load it, use the command LOAD "spread" (or LOAD ""), in accordance with chapter 20 of your Spectrum manual.

If the volume setting on your cassette player is suitable, the program should load first time. If you experience difficulty, consult chapter 6 of Sinclair's short introduction to the Spectrum first. It may help to try loading the program from a different cassette player. If still unsuccessful, please return your tape to MICROL, and we will gladly replace it.

Sometimes you find that your cassette player works properly with tapes which you produce yourself, but not with tapes produced on someone else's machine. In that case, you will need to borrow a cassette player, load your spreadsheet program, and then save it again using your own cassette player. That creates a copy which your normal cassette player can read.

The main program takes about 3 minutes to load. There is a shorter version on the same tape, without a built-in spreadsheet, which you may prefer to use when designing new spreadsheets from scratch.

### Copyright

Whilst on the subject of loading, your attention is drawn to the copyright notice on the cassette, which is strictly enforced. This states that you are only entitled to use this program on one single Spectrum computer. You are welcome to make as many back-up copies of the program as you need for use with that computer, but if you want to run it on other Spectrum computers, we ask you to buy a separate copy for each machine.

MICROL is prepared to grant block licences to colleges and firms wanting to copy this program for use with several machines. Please contact us for full details.

## Chapter 2 Fundamentals

### Looking at the screen

Load the main program, as explained in the last chapter, and you will find yourself looking at the first of our demonstration spreadsheets.

The top line on the screen is known as the contents line, since it normally contains information about the contents of a particular cell in the spreadsheet. More about that shortly.

At present, the contents line contains the program name and the MiCROL copyright notice. This will vanish as soon as you start using the program.

Immediately underneath is a framework, shaped like an upside down L, which is printed against a bright background. It contains a series of letters and numbers (currently A to D and 1 to 20), to identify the columns and rows into which the rest of the screen is divided.

Each row contains four cells (one for each column). Any cell can be identified by quoting its column letter and row number, in that order. For instance, the word "Rows" near the middle of the screen is situated in cell B5.

There is also a bright patch in cell A1, highlighting the letters "Demons". This is the cursor. It tells us that the computer is ready to change the information in this cell if we wish. As we are not ready for anything as rash as that (yet), let's start with a simpler experiment. To be on the safe side, we'll assume that you are a complete novice at computing and using spreadsheets - if you are already expert in either area, you may be able to skim through the manual quite quickly.

### Moving around the screen

Some of the keys on the top row of your keyboard are marked with arrows. We can use them to move the cursor round the screen.

Press the key with the figure 6 (and the downwards pointing arrow) on it, and see what happens. The cursor moves down to cell A2. The computer also makes a slight click, and the contents line changes.

Now hold the 6 key down and count 10 clicks. The cursor should come to rest on cell A12. Don't worry if you overshoot, since the next thing to do is to hold down the 7 key until the cursor is back at cell A1.

As you are doing this, keep an eye on the contents line. Notice how the contents line shows you what is in each cell as you move over it.

You may also have heard the quiet "bleep" that the machine makes if you press the 7 key when you are already in row 1 - this reminds you that you can't go any further.

Now try pressing the 5 and 8 keys (also marked with arrows). As you'd expect, they move the cursor to the left and right, along the line.

You'll find that you can't move leftwards from column A, but you can go right from column D. The results may surprise you! The screen goes blank, and the computer starts building up a fresh display at once, with columns B to E instead of A to D.

This is because there are four demonstration spreadsheets in this program. Each one fits neatly into a single screen, but if you move the cursor around in a trial and error fashion, you can end up with bits of two spreadsheets on the screen at the same time. This doesn't cause any damage, so feel free to experiment.

#### Turning the page (The GO TO command)

When you are ready, press the G key (using G or g, whichever suits you best). Then look at the foot of the screen.

A message has appeared (called a prompt), asking you where you want to go to. Type H1 (again, H1 or h1 will do), and press ENTER. The screen clears, and the computer starts building up the second demonstration spreadsheet (Bank Account). The last detail to appear on the screen is the cursor. Notice that it is on cell H1 - you have just seen how to operate the GO TO command, which takes you straight to a cell of your choice.

Don't stop to find out how the bank account works - we'll return to it later. Instead, press G again, enter A40 and press ENTER again. Once again, a new spreadsheet appears, this time a Journey Planner.

Finally, press G, but this time enter H by itself. The

computer assumes that you mean to stay on line 40 and moves the cursor to H40. The Cash Forecasts spreadsheet now appears on the screen.

This has just demonstrated one of the MiCROL spreadsheet's short cuts. In nearly every situation where you need to quote cell references, a partial reference will do just as well, if it is in the same column or row as the cursor is in at present. For instance, when the cursor is at H40, cell D means cell D40, and cell 19 means cell H19.

### Editing entries

We are now almost ready to return to the first spreadsheet. First, a couple of harmless experiments.

Press G and enter A99. The program won't like this, because cell A99 is not in use at present. But press ENTER all the same, and see what happens.

There is a quiet bleep, and an ERROR message appears at the foot of the screen. A moment later, the warning vanishes, and you can now try again.

If you are not very familiar with the Spectrum, the following drill may interest you. First press G and enter Z123. That's out of range, too. But this time, correct it to A1 before you press ENTER. Use the DELETE key at the top of the keyboard to rub out the offending 23 (don't forget to hold down CAPS SHIFT at the same time). Then press the 6 key once (still holding down CAPS SHIFT), to make the Sinclair entry cursor (the black flashing L) skip over the 1. Now press DELETE to get rid of the Z, then press A, so that the GOTO address is A1.

This is called editing an entry. It may sound complicated, but you quickly get accustomed to it. Whenever you have to type something and press ENTER afterwards, you can change your mind in this way at any time before you actually press ENTER. Useful to know!

Finally, remember that we all hit wrong keys at times. To err is human - but computers can be programmed to forgive, and the MiCROL program is particularly understanding. So don't worry. A warning usually appears on the screen when you make a mistake, and vanishes again after a second. As soon as it has gone, you can carry on where you left off. Chapter 8 explains what to do if anything worse than this happens.

## Chapter 3 Entering data (The Dimensions model)

### Introduction

This chapter uses the first demonstration spreadsheet (headed Dimensions) as an illustration. If you haven't already done so, load your spreadsheet program and check that the cursor is on cell A1. If necessary, use the GO TO command (see last chapter) to put it there.

Individual spreadsheets are usually known as models. The complete grid of rows and columns on which the models are created is called a matrix. Later in this manual you will see how to form a new matrix by using the NEW command.

This model calculates how many cells there are on a matrix of a given size, and how many screenfuls (or pages) of information it holds, assuming that there are 4 columns and 20 rows on each page.

The procedure is simple. You enter the number of columns and rows in columns A and B respectively. The computer calculates the number of cells and pages, and displays them in columns C and D. As you can see, five sets of dimensions have already been worked out, but there is still room for more. So let's see how many pages are needed to display a matrix with 20 rows and 25 columns.

### Entering a constant (The EXPRESSION command)

To enter the new data, move the cursor down to the first free row (cell A13). You can use either the cursor-down control (the 6 key) or the GO TO command for this purpose (as you're already in column A, the partial reference GO TO 13 will be enough).

Notice that when you arrive there, the contents line shows the cell reference (A13) and nothing else. To put the figure 20 in the cell, you need the EXPRESSION command, which works as follows:

- (1) Press E (or SPACE). The program replies by displaying expression: at the foot of the screen.
- (2) Type in 20 and press the ENTER key. The computer puts the figure 20 in cell A13, and displays 20 beside the cell reference in the contents line.

Now move across to B13 (by pressing the 8 key) and enter the figure 25 in this cell (ie, press E, type 25, press ENTER).



If you type in the wrong amount, but see the mistake in time, correct it by using the editing method given in the last chapter. If you only spot it after pressing ENTER, you'll need to use the EXPRESSION command again - the wrong figure disappears when you enter the correct amount.

### What's a block? (The RECALCULATE command)

Press C (the CALCULATE command). A brief message appears at the foot of the screen, asking you where you want to calculate from:. Type C and press ENTER. The word to: is now added to the message. This time, type D and press ENTER. You've just told the computer to work out the answers, and place them on the same line in columns C and D, using the formulas which are already stored in cells C13 and D13.

The figures 500 and 10 appear in columns C and D, one after the other, telling us that a 20 by 25 matrix contains 500 cells, and fills 10 pages.

When you entered C and D, you were actually defining a block of cells which needed to be calculated. The idea of a block is used in several commands, so let's look at it carefully.

A block is a rectangular box of cells, which is identified by naming the cells in two opposite corners (usually the top left and bottom right hand ones). For example, there is a block of blank cells in columns A and B (rows 14 to 20). This is normally called block A14 to B20, although the computer won't mind if you refer to it as B20 to A14 - or even A20 to B14 or B14 to A20! The two references defining a block are often called its range.

A block can consist of part or all of a single row or column. For instance, the first three headings on row 6 (from Columns to Cells) form the block AS to C5, whilst the figures in column D are block D7 to D20.

The smallest possible block is a single cell. There's a small catch here - a block reference must always contain two cell references, so the word Pages in cell D5 forms the block D5 to D5. And the largest possible block is the whole matrix, which reaches from A1 to H40 at present.

As explained in the last chapter, you can often use partial references. In fact, that is precisely what you did with the CALCULATE command. The cursor was already in row 13, so the computer interpreted C and D as C13 and D13.

If you press the ENTER key without giving any reference at all, it assumes you mean your current cell. So to D means the same as B13 to D13 (provided you press ENTER without touching the SPACE key, where the gap has been left). Pressing ENTER twice tells the computer to recalculate everything from your current cell to your current cell. And that is just one cell, of course.

Finally, the computer remembers the last range which you specified. Consequently, you can tell it to "use the same range as last time", without typing in the cell references again. This is done by typing \* in response to the "from:" question (and then pressing ENTER in the usual way). But be careful not to type \* in reply to the "to:" question - this stops the program at once.

### Practice

It is a good idea to spend some time practising at this stage, watching exactly what happens as the computer obeys the EXPRESSION and CALCULATE commands. Try entering some data of your choice in cells A14 and B14, then calculate the block B13 to D14 (using the partial references 13 to D, since you will have ended up at cell B14).

See how the contents line behaves during calculation. First, the information about the current cell vanishes. Then a series of cell references appears (B13, B14, C13, C14, etc), to tell you which particular cells have been calculated. At the same time, the revised figure for that cell appears on the spreadsheet, if any alteration is needed. When all the calculations have been finished, the machine bleeps to tell you that it is waiting for your command, and details of the current cell reappear on the contents line.

Feel free to experiment with these commands. You won't harm the spreadsheet unless you start entering data in columns C and D. In any case, it won't take long to reload the program if you hit disaster. With a little trial and error you will discover that:

1. It doesn't matter where your cursor is when you use the CALCULATE command - provided you give the right references, you can calculate ANY block in the matrix (even if it is not on your screen at present).
2. If you enter letters instead of numbers, or use invalid references like AP2 or Z123, the program usually stops and has to be restarted by entering GOTO go (see chapter 8). Sometimes it thinks that a letter is a number, since several letters are used as numbers inside the spreadsheet program - that causes confusion, so take care.

### Changing the formulas (The EXPRESSION and LABEL commands)

It is now time to meet the most important feature of this program. Move the cursor to cell D7, then look at the contents line.

Although the cell displays the figure 1, the contents line contains a long formula - so long, indeed, that it can't be shown in full. If you could see the missing details, it would read:

$$\text{INT} (.99+n(1,7)/4)*\text{INT} (.99+n(2,7)/20)$$

This is the formula for calculating the number of pages needed to display a given matrix, assuming that there are 4 columns and 20 lines per page.

Formulas like this are entered with the EXPRESSION command, so try one yourself. Press E (or SPACE) and type in this formula:

$$A7 / 4 * B7 / 20$$

This expression tells the computer to:

1. take the value in cell A7 and divide it by 4
2. multiply this by the value in cell B7 divided by 20.

It is best to type the whole formula without any gaps to save space in the computer's memory. However, you can leave gaps if you wish, provided you don't put a space between the letter and number in a cell reference. The cell references must be typed in full - partial references cannot be used in expressions. The mathematical symbols (/ for divide, \* for multiply) are found on the bottom row of the keyboard (press the SYMBOL SHIFT key at the same time).

The program stops to think for an moment or two after you press the ENTER key. Then it displays its own version of the formula on the contents line, and puts the figure 0.5 in cell D7, You'll find that the computer translates the formula into:

$$n(1,7) / 4 * n(2,7) / 20$$

As you can see, n(1,7) and n(2,7) is the program's internal way of referring to cells A8 and B8 - the n means a cell, and the column letters have been changed into column numbers, so the figures in brackets are the column and row references for each cell.

The figure of 0.5 in cell D7 has been obtained by using the current values of cells A7 and B7 in this formula.

Your new formula has calculated the exact fraction of a page covered by a matrix of 4 columns and 10 rows. The old formula, which was entered as:

$$\text{INT} (.99 + A8 / 4) * \text{INT} (.99 + B8 / 20)$$

worked out the number of whole pages which it occupies. Try entering it yourself.

INT tells the computer to take the integer value of the term which follows. It is entered in the Sinclair EXTENDED mode (first press both SHIFT keys, then the R key - see your Sinclair manual for full details). The two figures of .99 are needed because INT rounds any fractions downwards - this formula needs them to be rounded upwards.

Note that formulas can include any Sinclair mathematical functions (including RND, which gives random numbers). Again, see your Sinclair manual for full details of these functions and how to use them.

You have now finished with the Dimensions spreadsheet as far as this manual is concerned. So feel free to experiment with it trying new formulas, or working out the sizes of other matrices. It is interesting, for instance, to see whether a matrix with 20 columns and 25 rows needs the same number of pages as one with 25 columns and 20 rows. Or how many pages you can save by cutting out one column or one row,

Finally, the entries in rows 11 and 12 show you the largest matrices obtainable with a 48K Spectrum. It can handle up to 26 columns or 99 rows, but not both at the same time.

## Chapter 4 Entering, text (The Bank Account model)

### Introduction

This chapter shows how to keep a record of the money passing through your bank account.

Load the spreadsheet program, if it is not already running, then use the GO TO command to move to cell H1. The computer will then display the Bank Account spreadsheet for you.

As you can see, it only has room at present for 16 lines of entries. Instructions for creating a larger version (with over 200 lines) are given at the end of the chapter.

### Displaying text (The LABEL and WIDTH commands).

This is a very simple spreadsheet. When you pay out money by cheque, standing order, bank charges, etc, you record the amount in column F. When money is paid into your account, you show it in column G. The balance left in the account is calculated by the computer and displayed in column H, with a minus sign in front of it if it happens to be an overdraft.

Suppose the next transaction is a payment of £21.75 for petrol. Written details of each item are given in column E, so start by placing the cursor on cell E9. Then press L (the LABEL command, since all written notes are called labels in spreadsheet modelling). The word label: appears at the foot of the screen, and you can type in Petrol. Press ENTER, and the new label is displayed in cell E9. It also appears on the contents line, with an apostrophe in front to show that it is a label and not an expression.

You may type anything you wish in a label, including numbers, symbols, and graphics characters (but don't forget to go into the Spectrum GRAPHICS mode first, and switch back to the ordinary LETTERS mode afterwards - see Chapter 14 of your Spectrum manual).

You will notice that the columns on the spreadsheet are only 7 characters wide at present. This can be changed by the WIDTH command - just press W, and enter the column width you want. Try it out now, using a width of 10. Even a small increase in column width cuts down the number of columns you can show on the screen at once. So return to a width of 7.

Fortunately, you do not need to fit your labels exactly into the column width. Try using a longer label for cell E9 - for example, Petrol for trip to Southend. Only the first 7 letters are displayed on the worksheet, but the whole label is shown on the contents line. This can be quite handy on occasion. Another useful trick is to alter the column width to 30 when you want to read the complete labels, then change it back to 7 afterwards.

You may notice that entering a long label (or expression) can make the bottom part of the screen display vanish. That doesn't matter - it will reappear next time you use the WIDTH command, or do anything else that makes the screen redisplay itself.

To complete the entry in your bank account, move to cell F9, enter the sum of £21.75, then recalculate cell H9. You should end up with a balance of £473.25.

#### Clearing the bank account spreadsheet (The BLANK command)

Suppose that all the lines in this spreadsheet have been used. We need a method of clearing out the old entries and starting with a clean sheet.

The answer is to use the BLANK command, which wipes out everything in a given block of cells. Just press B, and when the prompt appears, enter E5 followed by G20. The unwanted details vanish in almost a flash.

The next step is a little tricky. We want the current bank balance (£473.25 at present) transferred up to cell H5, and all the cells from H6 to H20 set at zero. But we can't enter these figures directly into column H, since these cells contain the formulas for calculating the bank balances - which mustn't be changed.

The trick is to recalculate the H column from H5 to H20, then enter the current bank balance as cash received in cell G5, and finally recalculate cell H5 again. If the bank balance was an overdraft, we would enter it as a payment in F5 instead.

First make a note of the bank balance, then recalculate H5 to H20. This resets all the cells in question at zero. Next, take the cursor up to E5 and put in a suitable label - usually the date of the bank balance (say May 1). Then across to G5 and put in 473.25 as an expression. Finally, recalculate H5 only. You're now ready to start again.

### Creating a new spreadsheet (The NEW and REPLICATE commands)

The procedure described in the next few paragraphs is drastic. It destroys the Demonstration spreadsheets and creates an entirely new spreadsheet. If you want the old spreadsheets back after that, the only way is to reload the main program.

Before starting, take a careful look at the headings used in the present spreadsheet. Notice how each column has its own label (row 3), with a line of hyphens underneath (row 4). Notice also the title in row 1 (spread across two columns), the use of graphics characters in row 2 to give a bold line, and the formulas in column H. If you were working without a manual, you'd need to copy the key formulas onto a sheet of paper before going further.

Now press N (for NEW). The computer checks that you mean to destroy your existing matrix by asking 'yes' for new model:. If you enter anything except yes (which must not be in capitals), the computer ignores the NEW command completely. Check for yourself!

When you are ready to go on, press N again, and enter yes this time. The message changes to columns:. Enter the number of columns you want (which must not exceed 26) - 12 is probably the best number of columns for a bank account.

The computer replies by asking rows (<=77):. This means that you can have up to 77 rows in your matrix. Well, you need plenty of room for a bank account, so enter 77.

The computer now asks length 28 okay?. This tells you the amount of memory available for each cell in the matrix, and checks whether it is enough. A figure of 28 is the smallest possible, but quite adequate for your present purposes. Once again, enter yes. There is a slight pause, then a brand new spreadsheet appears on the screen, with every cell blank. If you try moving the cursor around, you'll find there are 12 columns and 77 rows, exactly as planned.

If you need more memory per cell, enter anything except yes, and the computer goes back to the very first question. You can then say how many columns and rows you want this time, remembering that at least one of these figures must be smaller than originally planned in order to give more memory for each cell.

To escape from this part of the program without losing your old spreadsheet, type no (or anything else, except yes) whenever the computer asks you to type yes. The questions at the foot of the screen vanish, and your spreadsheet is safe.

## Heading up your new model (The REPLICATE command)

You will presumably want to give the spreadsheet a special title, such as Thomas Adam's Bank Account. This can be entered in cells A1 to D1 (using the LABEL command). Use cells A2 to D2 as well if you are short of room. Remember that titles usually have to be divided up into sections, a few characters in each cell.

Next underline your title. Start at cell B2 (or B3 if you used row 2 for part of the title) and create a label with nothing except a line in it. If you want a bold line, go into graphics mode and press the 3 key seven times. Don't forget to then switch off the graphics mode afterwards, or the program may come to a stop and have to be restarted (see chapter 8).

Once you've got a line in one cell, it can be copied into other cells by the REPLICATE command. Press R, and the prompt replicate from: appears at the foot of the screen. Enter A followed by D (or omit the A completely, just pressing ENTER), and the computer will do the rest.

The REPLICATE command is another of those commands that do something to a block of cells. It can be used to copy expressions as well as labels - more about that soon.

Finally, enter the four column headings in the next line: Details, £ Paid, £ Recd, and £ Bal, and underline them, using the LABEL and REPLICATE commands once again.

## Replicating expressions

The final task is to put in the formulas for calculating the bank balances. There are only three of them - all the other balance cells are filled by replication.

Let's assume your first transaction will be shown in row 5. Enter the following formula in cell D5:

$$C5 - B5$$

In other words, the bank balance is anything you enter in the cash received cell C5, less anything you enter in the payments cell B5.

The formula for D6 is slightly longer, because the balance left over from D5 has to be taken into consideration. It is:

$$D5 + C6 - B6$$

Enter this using the EXPRESSION command, then copy it into cells D7 to D77 with the REPLICATE command.



This time, the REPLICATE command is more complicated. As usual, you press R and enter details of the block into which the formula must be copied (here 7 to 77). But after several seconds' thought, the computer will ask you modify reference 1 (D5) ?.

What it has done is to check through your formula, and found a reference to cell D5. It now wants you to say whether you want to keep D5 exactly as it is when it copies the formula, or whether it should modify the reference (making it D6 for the next cell, D7 for the following cell, and so on). You want the references to be modified, so enter y or just press ENTER. Any other reply, and the computer will keep the reference exactly as it is.

In fact, you will have to answer this question three times in all, since there are three separate cell references in your formula. The answer is y for yes in every case. The computer then enters all your formulas for you, quite automatically.

Columns A to D represent the first part of the bank account. When you have filled all 77 rows in these columns, you will need to go to columns E to H and continue with your entries. After that, you will move on to columns I to L.

Working out the first bank balance in columns H and L needs a special formula, as the previous balances are at the foot of columns D and H, not on the line above. Assuming that you don't repeat the headings displayed at the top of columns A to D, the first balance in column H will appear in H1. The formula for this cell is:

$$D77 + G1 - F1$$

and it should be copied into cell L1 as well, using the REPLICATE command and modifying all the references.

The final task is to fill in the remaining formulas for columns H and L. These can be obtained by replicating D6 twice, first into block H2 to H77, then into L2 to L77 (modifying all the references in the formula in both cases).

When you've done all that, the new spreadsheet is ready for use - and you should be reasonably familiar with the REPLICATE command.

Chapter 5 More advanced commands.  
(The Journey Planner model).

Introduction

It is now time to look at some of the more advanced commands in the MiCROL spreadsheet program. They will be illustrated by reference to a new model, although you will also need to be aware of them if you intend to put the Bank Account spreadsheet to practical use.

In addition to introducing new commands, this chapter also explains some of the main ideas involved in computer modelling.

If necessary, load the main spreadsheet program. Then move the cursor to A40. The computer now displays the Journey Planner spreadsheet, which is designed to help estimate the likely cost of a business or holiday trip, covering a period of up to 11 days.

Setting out a spreadsheet.

Let's start by looking carefully at how this spreadsheet is set out.

Every spreadsheet has to have some cells which are reserved for input data. It is important to put these cells at the top of the screen, or down the left hand side, since the program works downwards and from left to right as it performs its calculations.

This model assumes that you will travel by car, paying for petrol, meals, and overnight accommodation, for a period of up to 11 days. For longer trips, you would need to create a larger version of the spreadsheet (perhaps with some extra refinements like a column for expenses such as ferry fares or entertainment).

At the top of the screen, you enter the miles per gallon that you expect to get from your car, the price per gallon of fuel, and the amount you are allowing per day for meals and accommodation. These figures go into cells A23, A24, C23, and C24. Use litres and kilometres instead of gallons and miles if you wish.

The distances which you expect to cover each day are entered in cells A28 to A38. These should be based on data from a road atlas if possible, but there are bound to be days where you just have to guess at how far you expect to drive.

The program does the rest, working out the quantity of fuel you need per day, the daily costs, and the total cost for the whole trip.

The last few paragraphs illustrate two closely related points about computer modelling, and estimating in general. You have to make assumptions, and it is usually best to keep them simple.

For instance, this model is based on the simple assumption that the fuel consumption will be constant throughout the trip. If you want to, you can extend the model so that you grade each day's journey into motorway travelling, town travelling, etc, specifying a different fuel consumption for each class of driving. That would be good fun from a hobbyist viewpoint - but would it have much effect on the final estimate? In the end, it is up to the user to decide, but you may find that the extra accuracy just isn't worth the extra work.

### Inspecting formulas (The QUERY command).

Move your cursor to cell C28, press Q, and watch the contents line. Nothing happens at first. Then the existing contents line vanishes, and the following information appears in its place:

C28: A28/A23

You have just tried out the QUERY command, which displays the formula used in any given cell, in the form in which it was first typed, with the cell reference in front. If the cell happens to contain a label, the word label is displayed instead of a formula. If the cell is empty, the message blank cell appears instead.

This particular formula tells us that the fuel consumed in day 1 will be the day's mileage (A28) divided by the miles per gallon (A23). When you've finished looking at it, press any key (as instructed by the prompt), and the computer will be ready to receive your next instruction.

Now do the same with cell C29. This time the formula is A29/A23. Notice anything? Yes, the first term has been modified from A28 to A29, but the second term is still A23. When cell C28 was replicated to give cells C29 to C38, the first reference was modified, the second reference was not.

Finally, move to D28 and use the QUERY command again. D28 contains a long formula, which doesn't fit in on the contents line normally. However, the QUERY command enables you to inspect the whole of the formula, which is:

$$C23 + C24 + (C28 * A24)$$

This says that the daily cost consists of meals (C23), hotel (C24) and fuel costs (C28 \* A24). The brackets can be left out as far as the computer is concerned, but are helpful when humans read the formula.

Check the formula in D29, and you will find that only one reference in the above formula was modified during replication.

#### Adding things up (The SUM command).

The totals in row 40 are obtained by using the SUM command. To see how it works, move the cursor to A40 and blank out the cell (press B, then press ENTER twice).

Now press S (for SUM). This is yet another command which refers to a block of cells. So identify the cells you want added (A28 to A38) by entering 28 and 38. Back comes the original total of 1672, and the words sum: A28 to A38 appear on the contents line.

It is also possible to total several columns of figures at the same time, putting the answer anywhere in the matrix. To try this out, move the cursor to cell D21 (the only spare cell on the screen), and give the command SUM from A28 to C38. The answer won't mean anything, since it doesn't make sense to add miles, days, and gallons together. But the computer still does its sums correctly, giving a total of 1794.6. Don't forget to blank out this unwanted figure when you've finished.

#### A matter of accuracy (The FORMAT and UNIVERSAL FORMAT commands).

Columns C and D are rather untidy at present, since some of the figures include decimals, and some do not.

If you want, you can instruct the computer to remember the exact figure in each cell, but to display it as a round sum.

To do this, you use the FORMAT command (strictly, the LOCAL FORMAT command, since it only affects a specified block of cells. Press F, then enter 0 to show that you want the figures displayed to zero decimal places, followed by the range of cells in question (say C28 to D38). All the figures in that block of cells will be redisplayed, correctly rounded to the nearest whole number.

To prove that the computer still remembers the exact figures,

recalculate cells C40 and D40. The existing totals of 56.6 and 332.88 do not change.

The program can display anything from 0 to 6 places of decimals. Where a local format has been set, it is shown as a figure in square brackets at the end of the contents line for the cell in question. To get rid of a local format, use the FORMAT command, but enter D or d (for delete) instead of the number of decimal places, then enter the range of cells which you want cleared.

The program normally displays two places of decimals in any cell which has not been given a local format. However, this can be changed by using the UNIVERSAL FORMAT command U (known as the GLOBAL format command in many spreadsheet programs). Press U, enter the new format required (0 to 6), and the screen will be redisplayed with the new format.

Formats are important for neatness, and because they help to overcome overflow problems. Where a column is only 7 characters wide, the longest numbers which it can display are:

9,999,999	(0 decimals)
99,999.9	(1 decimal - remember the . is a character too)
9,999.99	(2 decimals)
999.999	(3 decimals)
99.9999	(4 decimals)
9.99999	(5 decimals)
.099999	(6 decimals - if the figure is, say, .999999, the computer insists on printing it as 0.999999, which can't fit into a 7 character space)

If the number is too long for the column, the computer prints > > > > instead, to warn you that there is a valid number in the cell which can't be displayed correctly. One remedy is to change the column, width (as explained in the last chapter), which often spoils the screen layout. Another is to change the format, either locally or overall, to cut down on the number of decimal places displayed. The choice is yours.

### Sensitivity

Finally, some more about the art of estimating. A computer bases its forecasts prepared on your assumptions about things like fuel costs and hotel/restaurant prices. What if these were understated by, say, 10%?

You can use the computer to find out. First change the fuel

consumption to 27 miles per gallon (roughly 10% more than the current estimate) and recalculate the cells from C28 to D40. Then put the fuel consumption back to 30 miles per gallon, change the hotel and meals costs to £19.25 and £3.85 per day (a rise of 10%), and recalculate C28 to D40 again.

You should find that the change in hotel and meals costs has far more effect on the total than the rise in petrol consumption. This suggests that if you want to improve the accuracy of your forecasts, it is better to spend time investigating hotel prices than checking your fuel consumption. It also gives you some idea of what a 10% rise in either of these costs means in money terms.

This type of experimenting is known as sensitivity analysis. It is an important technique for assessing the reliability of a forecast, and for finding out which figures you really need to worry about when making your estimates. There is no point in wasting time on figures which can't have a significant effect on the final answer.

Chapter 6 Business planning  
(The Cash Forecasts model).

Introduction

One of the most important uses of spreadsheet programs is in business planning. This chapter shows you a model for forecasting the movements in a firm's bank balance (and highlighting possible problems before they arise), which can easily be adapted for predicting your personal cash position.

When you've loaded the main program, move the cursor to cell H40 so as to display the model on the screen.

How to forecast cash flows

Cash movements are often forecast, on a monthly basis, for twelve months ahead. The model consequently allows for twelve periods of time, which you can interpret as weeks, months, or even years, if you prefer. There are various ways of making the forecasts. This model uses a simple, broad brush, approach which provides considerable insight into a small firm's cash problems, without too much detailed work.

There are five main factors to consider:

1. How much working capital is actually needed to run the business.

Working capital is the amount of cash tied up in stocks and debtors (ie, money owing to you by your customers), less your creditors (ie, money which you owe to your suppliers). This you must calculate and enter in cell F24. Notice the figure required is the amount of working capital you really need, so if for example you are overstocked at present, base your estimate on your normal stock levels.

2. How much income your business is making, before paying your overhead expenses.

In many businesses, this is the figure of gross profit which your accountant works out for you each year. You may prefer to call it your gross margin. Enter this in G24. If you prepare your cash forecasts on a monthly basis, divide your annual income by 12 before entering it.

3. What overheads have to be paid, in cash, out of this income. When adding up your overheads, do not allow anything for the

depreciation of your equipment, motor vehicles, etc. The reason is that depreciation is not a cash expense, and so does not affect the cash figures. Take your yearly overheads (ignoring depreciation) and divide by 12, entering the answer in H24.

4. The average growth rate which you expect to achieve during the period covered by the forecast (usually exactly one year). Here you want the real growth rate (ignoring price rises, which are handled separately). Again, divide by 12 to get a monthly figure. Then convert it into a growth factor (as explained below) and enter it in F25.

5. The average inflation rate which you expect to experience during that period.

This is bound to be a guess, since even experts get their inflation forecasts wrong. But it is a vital figure, and you can use sensitivity analysis later on (as explained in the last chapter) to see how your cash forecasts will be affected if you guess wrong. Divide the expected annual rate of inflation by 12, turn it into an inflation factor, and enter it in G25.

In working out the likely cash flows, the model makes the following assumptions:

a. If your business expands, your working capital has to expand at the same rate (eg, if you get 15% more income, you will need 15% more working capital).

b. Your working capital requirements are also affected by inflation (eg, if prices rise by 10%, you need 10% more working capital).

c. Your income will grow at the rate you forecast in F25. In addition, it also rises in line with inflation.

d. Your overheads are affected by inflation, but not by your growth rate - if the business expands by 15%, you will be able to manage with the same staff, the same premises, the same phone bills, etc. However, if you foresee any extra expenses, you can show them in the Extras column (E28 to E39), as explained below.

#### Percentages and factors

Some people get confused about percentages and factors. Feel free to skip this section if you're at home with them.



Suppose you have £240 on deposit at your bank, which earns 10% interest. The interest amounts to £24 and the bank balance consequently goes up to £264.

Mathematically speaking, it is easier to say that the interest rate is 0.1 (ie, 10% divided by 100), so that you can calculate the amount of interest by multiplying £240 by 0.1. It is also handy to say that your bank balance grows by a factor of 1.1 (ie,  $0.1 + 1$ ), so that you can work out your new bank balance by multiplying £240 by 1.1.

Using this idea, we can talk about a firm's growth factor or the expected inflation factor.

Suppose you plan to sell 12% more goods next year than you did this year. Your annual growth rate is 0.12 (ie, 12% divided by 100), and your annual growth factor is 1.12. If your sales were £200,000 this year, they would rise to £200,000 per annum multiplied by the growth factor of 1.12, in a year's time (thus making £224,000 in all).

A growth rate of 12% per year is roughly the same as 1% per month (for mathematical reasons, it is not exactly the same, but dividing by 12 is a fair approximation). So the monthly growth rate can be estimated as 0.01 (ie, 12% divided by 12 divided by 100), and the monthly growth factor as 1.01 (ie,  $0.01 + 1$ ).

Move your cursor to cell F25 and check the contents line. Notice how the monthly growth factor was worked out and entered at the same time by keying the expression  $12 / 100 / 12 + 1$ . It's a useful dodge.

At present the model is using an inflation factor of 1. This implies a zero inflation rate, which is somewhat optimistic under present day conditions. Later in this chapter, you will have the opportunity of seeing what happens to the cash forecasts when you allow for inflation. The results are quite disturbing.

### Calculating the cash flows

The amount of money which is expected to flow into the firm's bank each month is worked out in column H, using any supplementary data entered in column E, and the figures which have previously been worked out in column G.

Column E contains details of any special receipts or payments which are expected in the coming months. Cell E34 shows that

£6,000 is due to be spent in period 7 (perhaps the firm is going to buy a new van). This column can also be used to show seasonal variations in the level of income and overheads - if the firm expects to make extra money from the summer sales in period 8, for instance, the sum in question should be entered (as a positive figure) in E35.

Column G shows the amount of working capital which the firm is expected to need at the end of each period. This is calculated by assuming that the working capital expands at the same rate as the firm's income, also increasing in line with inflation. The formula for cell G28 is consequently:

$$F24 * F25 * G25$$

(check this for yourself with the QUERY command). For cell G29 this becomes:

$$G28 * F25 * G25$$

and that expression is replicated for cells G30 to G39 (modifying the first term only).

The important point here is that the increases in working capital have to be paid for out of the firm's income. So, comparing C28 with F24, you can see that the firm will have to spend £200 cash increasing its working capital in period 1. In the following period, the amount will be £202 (C29 - C28), and this rises steadily throughout the year to £224 in period 12.

The net amount of cash flowing into the business is shown in column H. If you use the QUERY command, you will see that the formula for H29 is a very long one:

$$G24 * (F25 * G25) ^ F29 - H24 * G25 ^ F29 + G28 - G29 + E29$$

Taken step by step, however, it is fairly easy to follow.

$G24 * (F25 * G25) ^ F29$  is the amount of income expected for the period. The real growth factor (F25) is multiplied by the inflation factor (G25) to estimate the overall increase in income per period. This is multiplied by itself the number of times indicated by F29 to estimate the overall increase since the start of the year (see chapter 10 of the Sinclair manual if you want to know more about the ^ symbol). Finally,  $(F25 * G25) ^ F29$  is multiplied by G24 to complete the income forecast.

$H24 * G25 ^ F29$  is the amount likely to be paid in overheads for the period. This is worked out in the same way as the income, but the growth factor is ignored (see assumption d. above).

$G28 - G29$  is the amount of money which will have to be invested in additional working capital, and E29 represents any extras from column E.

The H29 formula is replicated into H30 to H39, modifying the terms F29 (which occurs twice), G28, G29, and E29. A shorter version can be used for the first period, as you can see by moving to cell H28 and pressing Q. Finally, the total of column H is displayed in H40.

### Using the model.

At present, the spreadsheet forecasts that the firm's bank balance will increase by over £29,000 if the assumptions made in the model are correct. In particular, the firm must be able to increase its volume of trade by 12% during the year without extra overheads.

Let's check through the first few figures to make sure the model is working correctly. No point in blindly following the formulas used in the model, since they might be wrong - instead, think the problem through in a slightly different way.

Without any growth in trade, the cash inflow would be £1,000 a month (G24 - H24). With the proposed growth, the income for period 1 rises by £320 to £32,320 (G25 times F25), so the cash coming in becomes £1,320. However, an extra £200 is invested in working capital, which rises by 1% to 20,200, leaving a net inflow of cash of £1,120 as in H28. OK so far.

In period 2, the income rises by 1% again, from £32,320 to £32,643.20. So the cash coming in rises by £323.20 to £1,643.20, of which £202 is needed for extra working capital. This leaves a net inflow of £1,441.20, which is displayed as £1,441 since the local format is zero decimal places. OK again.

All the figures can be checked in this way. However, that doesn't prove that the model deals with inflation correctly, since it currently assumes zero inflation. So enter the factor for 12% annual inflation in G25 (keying in  $12 / 100 / 12 + 1$ , as in F25). Then recalculate G28 to H40.

The net cash inflow for period 1 falls dramatically. The new figure of £931 consists of the previous forecast of £1,120, plus the effect of a 1% price rise on three items. There is £323.20 more income, £310 more overheads and £202 more to be invested in working capital. And  $£1,120 + £323.20 - £310 - £202$  equals £931.20, again confirming the model.

Enough of the mathematics. The important point is that the total cash inflow for the whole year has hardly risen at all -

certainly not enough to compensate for the falling spending power of money. So the business is actually going to be worse off in real terms, despite the fact that its income is rising in line with inflation. The reason for this is the massive increase in working capital needed to keep pace with inflation.

If you are in business, try running the model with data from your firm. The results may surprise you, and even alarm you. One of the more interesting experiments is to assume a growth rate of say 12% per annum and a zero inflation rate, then switch the figures round, and assume zero growth coupled with 12% inflation. You'll find that you need to invest just as much money in working capital in both cases. When this is due to inflation rather than real growth, the cash inflow suffers badly, and you see why it is so difficult for firms to cope with rising prices.

### Over to you

This manual can only provide a brief introduction to the art of creating and using spreadsheet models. The previous chapters have shown you how to use the program, and given you some examples to follow. There are several points worth stressing.

Keep your models simple. Decide what factors are important in a given problem, and base your model on them, ignoring the frills. Then experiment, changing your key assumptions until you have a clear idea of how each factor affects you, and can identify the points which matter most in your particular situation.

Take care over screen design. Plan your models so that input data is entered at the top or left hand part of the spreadsheet. Let your models cover several screens if you wish, but try to get the main answers onto a single screen if possible.

Above all, use common sense. A computer can't guarantee what will happen in the future. It is up to you to decide whether the assumptions and simplifications you are using make sense, and how much reliance you can put on the computer's predictions.

When you've finished reading the manual, you may care to adapt this model so that it forecasts your personal finances for the next 12 years. The key input data will consist of your income and living expenses, the growth factor for your income (based on your present pay scale), and the inflation factor. You may want to use separate screens for each wage earner in the family, and introduce special columns for items like the cost of caring for elderly dependents.

Chapter 7 Some additional commands  
for calculating, printing, and saving.

Introduction

You have now met most of the commands for constructing and using spreadsheets. However, there is a group of commands which are best explained without the demonstration spreadsheets. These are dealt with in this chapter.

If you need to reload the spreadsheet program, use the short version of the program to save time.

More about calculating

The first step is to create a small experimental spreadsheet with 4 columns and 40 rows, using the NEW command. Then enter the following headings and expressions in rows 1 and 2, respectively:

Mileage	Mpg	mpg	Gallons
600	30	1.8	a2/b2

As you can see, this is a simplified version of the Journey Planner model which you met in chapter 5. We're going to use it for an experiment, so enter the label Cost in cell A4, with the formula c2/d2 under it, and REPLICATE these entries in cells D4 and D5, respectively. You'll find that the computer waits about 20 seconds before replicating the expression in A5 - we'll explain why in the next chapter, and show you how to speed things up if you wish.

Cell D2 should now contain the figure 20, representing the quantity of fuel needed for a journey of 600 miles. The cost of this amount of fuel appears in cells A5 and D5, and should be exactly 36. Make sure that you have got these figures on your spreadsheet - if not, check your data and expressions, and correct any mistakes which may have occurred.

Trials (and errors) (The CALCULATE ACROSS command)

By now, you are probably wondering why we've worked out the cost of the fuel twice. Well, move the cursor back to cell A2 and enter 500. We're going to make an experiment.

Start by calculating the whole matrix, using a new command. Press X (which stands for ACROSS the whole matrix), and the value

of every cell will be calculated for you. Specialists might call this a "global" calculation, but we prefer to remember it as the CALCULATE ACROSS command.

Next check the figures in A5 and D5. D5 shows 30 (the right answer), but A5 should still contain 36 (the original answer), which is now distinctly out of date. To get the right answer in A5, you'll need to press X a second time - try it for yourself.

#### Order of calculation (The ORDER command).

Now for the explanations. Press X again and watch the cell references shown at the left hand end of the contents line.

You will recall that these tell you which cells are being calculated. The sequence is A2, A5, B2, C2, D2, D5. In other words, the computer is working through the spreadsheet column by column starting with A and moving steadily across to Z (if you happen to be using all 26 columns).

Our problem is caused by the order in which the calculations are performed. The value of A5 depends on the value of D2. We've made the mistake of telling the computer to work out A5 before it checks D2. So we are bound to get the wrong answer in A5 whenever D2 changes during the course of a calculation.

It is vital to be aware of this factor when you are designing spreadsheets. If you let your program make its calculations in the wrong order, you can guarantee trouble.

Sometimes you can get round this problem by changing the sequence in which calculations are made. To see how this works, press the letter O (which is the ORDER command) and watch the contents line. The words:

"recalculation order now rows"

are displayed for about half a second, then the contents line returns to its normal display. Now press X, still watching the contents line, and you will see that the calculations are made in the sequence A2, B2, C3, D2, A5, D5 (ie, row by row).

To go back to the original order, press O again. This time the message is:

"recalculation order now columns"

and that's all there is to it.

If you experiment with a number of different mileages in A2, changing the order of calculation from time to time, you'll find

that cells A5 and D5 both give the right answer, first time, when the calculations are made row by row. This is because A5 and D5 both come after D2 in the sequence A2, B2, C2, D2, A5, D5.

When you've done that, rearrange the model so that the gallons required are calculated at D7 instead of D2 (remembering to alter the expressions in A5 and D5 accordingly), and repeat your experiments. This time, you'll find that neither A5 nor D5 give the right answer first time, whichever order of calculation is used. That's because D7 is the very last cell in the matrix to be calculated, whichever sequence is followed.

So the moral is this: always check that the calculations on your spreadsheets are made in the right order. The ORDER command lets you change the sequence of calculation if you wish, but it cannot prevent wrong answers appearing on the screen if you plan your spreadsheets incorrectly.

#### Printing out a spreadsheet (The TYPE and PRINT commands).

If you own a ZX printer, you may want to print out a copy of part, or all, of your spreadsheet - complete, perhaps, with an extra heading or footnote which doesn't appear on the spreadsheet itself.

The main PRINT command is the letter P. When you press it, the computer replies with the prompts print from: and to:. Enter the block of cells which you want printed out (or just a \*, if you wish to repeat the same block as last time). The required section of the spreadsheet will then be printed out.

There are two important points to note.

First, only the spreadsheet is printed. The computer does not print out the contents line at the top of the screen, nor the column and row references (A...Z and 1...99) in the framework around the screen.

Second, wider spreadsheets must be printed in sections, since the printer paper is only 32 characters across. As a result, the printed version may consist of several strips of paper, which have to be sellotaped together before you can view the full spreadsheet at one go.

If your spreadsheet columns are 7 characters wide, and there are more than 4 of them in the spreadsheet, the first strip of paper will contain columns A to D, plus the first four characters

of column E. The second strip will contain the last 3 characters from column E, plus columns F G H and I, and one character from column J. Get the idea?

If you want to avoid splitting columns in this way, you must either print blocks of four columns at a time, or reset the column width to 8 before printing. Don't forget, however, that resetting the column width may alter the appearance of the spreadsheet.

There are no restrictions on the length of these strips. If your spreadsheet has 99 rows, for example, and you decide to print it out in full, each strip will be 99 rows long (roughly 12 inches of paper).

Headings, footnotes, comments, etc, can be inserted before or after each block is printed. Press T (the TYPE command), and a prompt appears at the foot of the screen inviting you to type: whatever you want. Enter your note, then press ENTER, and your note will be printed out exactly as shown on the screen.

If you really want to, you can actually type 22 lines of notes before pressing ENTER. The catch is that the computer gets sluggish after the first 5 or 6 lines.

You will also spoil the screen display if you type more than one line of notes. This doesn't matter very much, since the computer redispays the spreadsheet whenever you use the WIDTH or GO TO commands. In fact, you can trick the computer into doing this at any time by pressing W and telling it to change the width to the existing width (eg, from 7 to 7). Try it for yourself.

#### Saving your spreadsheets (The SAVE command)

To save a copy of this program on cassette, complete with your current spreadsheet, press \$ (the SAVE command). Remember that the \$ key is the same as the 4 key, but you need to hold down the SYMBOL SHIFT whilst pressing it.

The computer replies name of file:. Type in the name by which you want this version of the program identified on the cassette, and press ENTER. If the name is more than 10 characters long, it ignores the excess.

The machine, replies with its usual message start tape, then press any key. Don't forget to disconnect the ear lead on the cassette recorder, and rewind the tape if necessary, before recording.



The program does not allow you to verify that the spreadsheet was saved correctly. Most people prefer to save the spreadsheet twice, which is quicker and easier than verification. If you particularly want to verify a recording, use the procedure described in the next chapter.

The spreadsheet which you have saved runs automatically when you reload it. Do not use the Spectrum's RUN command, as this deletes your existing spreadsheet, and replaces it with a blank one.

The saving process takes almost exactly three minutes. You can stop this by pressing BREAK if you need to, and then restart the program by typing GO TO go, followed by ENTER.

## Chapter 8 How to deal with a crisis

### Introduction

Everyone meets a crisis occasionally when using a computer. This chapter explains how to deal with the main types of crisis which you may encounter when using the MiCROL spreadsheet program. The first half of the chapter is compulsory reading for all users. Some of the later paragraphs deal with advanced points, and may be skipped if you wish.

The golden rule is not to panic if something goes wrong. You can't damage the computer by pressing the wrong key, so relax and feel free to experiment. The worst you can normally do is to lose the spreadsheet in the computer's memory. So it is always a good idea to save your spreadsheets on cassette as soon as you finish them, and before you start your experiments. Then you can easily reload them if something goes wrong.

### How to stop your program - and start it again

If you press a wrong key when the computer is expecting a command, it beeps and displays the message:

ERROR - undefined command

for about a second. Then the message disappears, and you can continue as normal.

That's the theory of it. But now suppose you hit the BREAK key by mistake (ie, by touching the SPACE key when the CAPS SHIFT key is held down - try for yourself if the program is running).

In this case, the computer does not beep, and instead of the usual ERROR. - undefined command message, it reports:

L BREAK into program, 60:1

(or something similar).

Another equally effective way of upsetting the computer to use a nonsense formula as an expression. Try something like 3-\*4, or 167 - AAA, and it stops with a message like:

C Nonsense in BASIC, 1010: 1

Even entering a space instead of a formula or number counts as nonsense - but somewhat surprisingly, it accepts 3+4 or 3+-4, since these expressions do have a mathematical meaning.

You may get the same reaction if you give a nonsense formula

as a cell reference. Alternatively, you may be told:

```
2 Variable not found, 1788: 2
```

or some other obscure fact. The choice of complaints seems almost endless.

If you are accustomed to programming the Spectrum yourself, you will know that such messages mean that the spreadsheet program has stopped running. The normal spreadsheet commands will not work until it has started running again, so your next commands) must be in the BASIC computer language.

You've got three options. The easiest is to press the C key. The computer replies CONTINUE, you press ENTER, and if all is well, the spreadsheet program resumes work where it left off.

If you are not so lucky, the computer simply repeats the message it gave you previously. This means that it can't restart the program where it stopped, and needs another starting place. So press the G key. This time, the computer replies GO TO, and waits for you to type in something extra. Type GO (or go), press ENTER, and once again, all should be well.

By now, however, you may have pressed ENTER once too often by mistake. If you do not understand programming, the results may scare you, since the screen is filled with a stream of non-stop information. This is actually the MiCROL spreadsheet program. To stop it, press the SPACE key firmly. Then press G, followed by RS (or rs) and ENTER. The computer clears the screen, redisplayes the spreadsheet, and prints the MiCROL caption on the contents line. Everything is now ready for you to continue.

- To summarise, the three ways of restarting the program are:
- a) Press C and ENTER (the CONTINUE command in BASIC),
  - b) Press G, type go, press ENTER (the GO TO GO command in BASIC)
  - c) Press G, type rs, press ENTER (the GO TO RS command in BASIC)

GO is the place in the program where the machine waits for your next command (ie, your "go"). RS is the place in the program where it restarts the screen display ("rs" stands for ReStart).

#### How to kill the program - and what to do then

If you follow the above rules, you should not have any difficulty in restarting the program whenever the need arises.

However, if you press the A key after the program stops, and then press ENTER, the screen will go black, then white again, and

you will have killed your program completely.

If you press R (or X) and ENTER, the program will stay in the machine, but you will have killed your spreadsheet.

In both cases, the answer is to reload the program from your cassette. It is unusual to press A, R or X by mistake, so don't have nightmares about it, but do make a practice of saving every spreadsheet on cassette, just in case...

### Dealing with overflows.

Try entering a really large number as an expression on the spreadsheet - say 1,234,567,890,000. If your column width is 7 (the normal width), then the number is too long to fit into the column. The computer tells you so by displaying

> > > > > >

in the cell in question. As explained in chapter 5, this is the OVERFLOW symbol.

Now press W and change the column width to (say) 15. This makes enough room for displaying the number, but instead of the 1234567890000 which you might expect, a figure of 1.23456789E+12 appears. This is mathematical shorthand for 1.23456789 times 1,000,000,000,000 (see Spectrum manual, chapter 7). The Spectrum uses this shorthand for all numbers of 100,000,000 or more.

Unless you have a very special reason for using wide columns and mathematical shorthand, you will want to keep your numbers short enough to fit into a column width of somewhere between 7 and 10 characters (7 gives a 4 column display on the screen, 10 gives a 3 column display). There are two tricks which may help you. First, use the format commands to restrict the number of decimals displayed - it is quite safe to cut out the decimals entirely for display purposes, since the full number is stored accurately in the computer's memory. Secondly, scale down your figures if need be. Work in thousands or millions rather than units (but remember to make the necessary corrections to your formulas - 2,000 multiplied by 2,000 is 4,000,000, not 4,000!).

### Faster replication, faster queries.

We thoroughly recommend you to keep your spreadsheets as small as possible. Small, well planned, models are more effective than large, spidery, models. They're easier to use, and often more reliable, since you are less likely to get lost in your own

calculations. And another advantage is that you can use longer formulas in your expressions - as explained in chapter 4, there's a direct trade off between expression length and matrix size. So plan ahead, and use the NEW command to tailor the matrix to your real requirements.

However, you may run into a different snag if you work with very long expression lengths. They take ages to REPLICATE and QUERY, since the computer stops to check every possible character in the whole expression - not just the ones you actually use. That's fair enough if you really need long formulas, but annoying otherwise.

As a general rule, do not use expression lengths in excess of about 100 if you want reasonably fast replication. If your matrix has less than say 250 cells (roughly a 4 by 60 matrix), then you may be glad to shorten the expression length artificially.

The method is as follows. First use the NEW command to set up the matrix. Then press BREAK (ie, CAPS SHIFT and SPACE) to stop the program. Now press L (the BASIC LET command), and type:

```
len = 100
```

Press ENTER, and the computer replies:

```
0 OK, 0: 1
```

to show that it understands. Finally, press G (the BASIC GO TO command), type 23, and press ENTER to restart the program.

This must be done before you start your spreadsheet, since any existing data in the matrix is cleared out as soon as you give the BASIC command GO TO 23.

You may use any expression length between 28 and the amount originally fixed by the NEW command. Just give a different value to len in your LET statement.

If the expression length fixed by the NEW command is too short, you may be able to increase it slightly by following this procedure. But that's at your own risk - don't be surprised if the program refuses to work properly afterwards.

#### More about saving spreadsheets

If you really want to verify that your spreadsheets are saved properly, do not use the SAVE command provided in the program. Instead, use the BREAK key to stop the program, and follow the normal Sinclair saving and verifying routine explained in chapter 20 of your Sinclair manual.

To make the program run automatically on loading, use the SAVE statement:

```
SAVE "program" LINE rs
```

where "program" represents the name under which you want it filed, and rs is the place at which the program must be restarted on loading. (Yes, this is the same starting place as used with the GO TO rs command).

If you wish, you can save the spreadsheet as a data file, without saving the program as well. Again, break into the program and follow the instructions in chapter 20 of the Sinclair manual. You will need the statements:

```
SAVE "matrix" DATA t$()
```

to save the spreadsheet, and

```
LOAD "matrix" DATA t$()
```

to reload it. You can use a name of your own choice instead of the word matrix, but always refer to t\$(). This method saves your numbers and formulae, but not the formats created by the F command, which are kept in another array, f\$(). Save and load this as well if necessary.

In addition, after reloading the model, you MUST reset two BASIC variables before restarting the program. The variables in question are called lastc and lastl, and they represent the number of columns and rows in the spreadsheet. If, for example, you are dealing with an 8 by 40 matrix, you will need the following BASIC statements:

```
LET lastc = 8
```

```
LET lastl = 40
```

Seeing you MUST remember both figures, it is a good idea to include them in the name under which you save the spreadsheet (eg, "matrix8x40"). It may look untidy, but it does save trouble.

Finally, bear in mind that the name must not be more than 10 characters long, including these figures. Names like "matrix8x40" or "matrix8.40" are in order, but "matrix8by40" or "matrix 8x40" are both too long.

## Chapter 9 Final reminders and suggestions

### Introduction

It may take you quite a long time to plan your first few spreadsheets, and get them looking as neat as you want. After that, you'll start to develop a feel for spreadsheet design, and you'll find yourself working much faster.

In the meantime, here are some reminders and suggestions which may help to make life easier for you.

### Crisp models

Once again, remember to keep the spreadsheets as compact as possible.

Try to reserve the left hand column, and possibly one or two rows near the top of the spreadsheet, for keying in any data you need. We were able to do this in 3 out of our 4 demonstration spreadsheets. This helps the user to work quickly, and lessens the risk of spoiling the model by entering data in a cell which already contains a formula.

Use symbols like "-----", "=====", "<=", and ">=", to underline your headings or to point to important figures in the spreadsheet. Experiment with the Spectrum's built-in graphics characters, which can be used to give a very bold line, and other dramatic visual effects.

Try designing special characters of your own, as explained in chapter 14 of your Spectrum manual. Then save these characters as bytes (Spectrum manual, chapter 20), using the BASIC command:

```
SAVE "graphics" CODE USR "a", 1336
```

Next time you use the spreadsheet program, reload the special characters first with the following command:

```
LOAD "graphics" CODE USR "a"
```

They will then remain in the machine until you next unplug it, and can be used in the spreadsheet itself.

If you want help with designing special characters, there is a short WORKBOX program on MiCROL's USE AND LEARN tape, which enables you to design characters on your screen. It also tells you what data you need to enter in a program of your own to set up the character(s) you design.

Remember, however, that your special characters will be replaced by ordinary letters of the alphabet if you load a spreadsheet containing special characters without loading those characters into your machine first.

### Use the Spectrum mathematical facilities to the full

Remember that you can use any of the Spectrum's mathematical functions in your spreadsheet formulas. Many of us get by quite well with +, -, /, and \*, but it is often handy to use symbols like SQR (for square root - obtained by pressing CAPS SHIFT and SYMBOL SHIFT together, followed by H), or INT (for integer - press CAPS SHIFT and SYMBOL SHIFT, then R). See chapters 9 and 10 of the Spectrum manual for more details.

Another handy symbol is RND (for random number - press CAPS SHIFT and SYMBOL SHIFT, then T). Next time you have an empty spreadsheet, try this experiment. Enter the EXPRESSION RND in cell A1. Take care - you want the RND symbol, as explained at the start of this paragraph, not the letters R N D. Then REPLICATE from A1 to D20.

Your screen will fill up with a series of random numbers between 0 and 1. Press X, and you'll get different random numbers. Press X again, and yet another screenful of numbers appears. And so on...for ever.

Now suppose you want to design a spreadsheet in which the computer chooses a whole number for you between 150 and 250, for use in some other calculation. You can get it to do this quite easily. The formula is:

$$150 + \text{INT} (\text{RND} * 100)$$

If you enter this in A1, and REPLICATE it from A1 to D20, you'll see 80 examples of the numbers which the computer might select.

Random numbers are often used nowadays to help us explore (or simulate) business problems. You could find them handy in some of your own spreadsheets.

### Why not choose your own commands?

Finally, you may end up wishing that the spreadsheet program used different keys for certain commands, or even included some extra command (like a COPY command) which we've not given you.

If you can cope with BASIC programming, feel free to adapt



the program accordingly. Use the BREAK key to stop the program, press K (the Spectrum LIST key), type in the word main, and press ENTER, and press SPACE to stop the listing.

You'll find yourself looking at the part of the spreadsheet program, starting at line 1500, where the computer interprets the keys which you press (identified as k\$). If you add the following line:

```
1552 IF k$="Z" THEN COPY: GOTO m2
```

and restart program again, the computer will print out an exact copy of the screen every time you press the Z key (provided the printer is attached!).

If you are accustomed to pressing CAPS SHIFT whenever you want to use the cursor control keys, you may be irritated by the way that the program always treats this as an error. In that case, edit lines 1542 to 1548, replacing the terms "7", "6", "8", and "5" with CHR\$ 11, CHR\$ 10, CHR\$ 9, and CHR\$ 8, respectively (without any quotation marks, please). Start the program, and the program will now complain if you fail to use the CAPS SHIFT when pressing a cursor key.

This section of the spreadsheet program goes as far as line 1593. It is safe to replace any of the letters shown in quotes by another capital letter (provided the new letter is not used elsewhere in this part of the program). The new letter then becomes your private code for the command which is usually denoted by the old letter. So if you'd like to use 1 as a GO TO command (instead of the normal G), change "G" to "1" in line 1565.

## Security

Changes like these can also be useful for security reasons. Whilst they won't defeat a competent BASIC programmer, they make it harder for someone to look at one of your spreadsheets if you don't want them to.

Suppose, for example, that you'd like to be able to kill the spreadsheet program in a hurry if someone comes into your office whilst you are working on confidential information. Adding one or more of the following lines to your program will do the trick:

```
1576 IF k$="K" THEN NEW
1577 IF k$="A" THEN GOTO 23
1578 IF k$="V" THEN CLS
```

The word NEW in line 1576 is the symbol obtained by pressing the A key when using BASIC - this line means that you can KILL the

entire program whilst it is actually running, by pressing K when it expects a command. Then no one can see your confidential data without first reloading the program from cassette.

Line 1577 is slightly less drastic - it kills your current spreadsheet when you press A, but leaves the program intact.

Line 1578 is even milder. CLS is the symbol obtained by pressing V in BASIC. If you have this line in your program, and press V when it is running, the screen goes blank and the computer complains that you have given an invalid command. However, the spreadsheet is still in its memory, and can be redisplayed by using the WIDTH command (see in the section on the TYPE command in chapter 7).

Other safety devices are possible. For instance, the line:  
1579 IF k\$-"y" THEN PAPER 0: BORDER 0: GO TO rs  
is quite effective when the Y key is pressed. The only way out of the fog which this causes is to stop the program and enter:

PAPER 7: BORDER 7: GOTO rs

If you try it, remember that PAPER and BORDER are single symbols (see Spectrum manual, chapter 16, for more details).

One or two traps like these can help to protect your program against people who try using it by trial and error without your permission. But don't go overboard with them - the program uses a lot of memory already, and you might run out of space.

### Postscript (The spreadsheet story).

You may be interested to read a little about the origin of spreadsheets.

The word "spreadsheet" is an American term, and seems to have been invented during the 20th century. It refers to the large sheets of paper (also called "worksheets") on which accountants set out complex financial calculations.

Although the word is new, the idea of displaying calculations in this way is much older, dating back at least to the time of Edward Thomas Jones, who worked as an accountant in Bristol at the end of the 18th century.

#### The Balance Book.

Jones, the eldest of seven children, was orphaned at the age of 14. Shortly afterwards he was apprenticed to one of Bristol's leading accountants. On completing his training, he became a freeman of the city, and his name can still be found in various documents preserved at Bristol Record Office.

One of the main problems in 18th century accounting was the lack of an efficient check on the accuracy of accounts. Edward Jones's great achievement was to invent a form of batch control which virtually eliminated the risk of error in a firm's account books. He called it The English System of Book-keeping, patented it, and published a best selling book on the subject in 1796.

To obtain his control totals, Jones first tried inserting extra columns in the account books themselves. This method was too cumbersome, and little used in practice. However, by 1820 he had found a better system. Instead of putting extra columns in the account books, he worked out the control figures on large sheets of paper. These were identical to modern spreadsheets, and could easily have been prepared by the MiCROL program if computers had been invented in those days.

Jones used to sell the pre-printed spreadsheets in slim volumes called balance books. One of his first customers was the famous East India Company. The system was later adopted by many textile firms, including John Foster and Son, owners of Black Dyke Mills (and founders of one of the world's finest brass bands).

The importance of balance books is shown by the fact that

they were still widely used more than 60 years after Jones's death (see, for instance, George Pepler Norton's influential book on Textile Manufacturers' Bookkeeping, published in 1889).

### The spreadsheet in America

These ideas were known in the United States, partly as a result of Jones's writings, and partly through the efforts of his son and grandson, who built up a flourishing accounting practice with offices in England, Ireland and America.

Consequently, it is not surprising to find that spreadsheets were featured in The New Bryant and Stratton Counting House Book-keeping, published in New York in 1878.

The interesting point about this book is that it contained two types of spreadsheet. One was similar to Jones's original balance book, the other resembled the sort of worksheet which a modern accountant might use to calculate a firm's profit and financial position. Neither was actually called a spreadsheet, but together they provide concrete evidence of the way in which the spreadsheet idea was developing.

During the first part of the 20th century, accountants found it necessary to take more and more interest in financial planning. This made them increasingly aware of the value of spreadsheets for setting out their calculations, although they still had to do their sums the hard way, by hand or with a calculator.

The invention of computers did not help very much, since the early programs were costly to write, and lacked the flexibility of a handwritten spreadsheet. So accountants continued to use pencils, rubbers, and calculators for much of their work.

Towards the end of the 1970s, however, two important things happened. The first was the development of the personal computer. The second was a stroke of genius by two American programmers, Dan Bricklin and Bob Frankston, who spotted that it is possible to create electronic spreadsheets on the new computers. And so we've reached the point at last where nearly everyone who works with spreadsheets has invested in a computer and spreadsheet program.

It would be hard to exaggerate the importance of their work. They did for computing what Henry Ford did for the motor car. MICROL is proud to follow in their steps by offering you what we believe to be the most elegant and easy-to-use spreadsheet program currently available in its price range.

## SUMMARY OF COMMANDS.

To move cursor: use the ARROW KEYS (5, 6, 7, and 8), unshifted

To. restart program (after errors or deliberate use of BREAK key): use the BASIC commands, CONTINUE, GO TO go, or GO TO rs

<u>The spreadsheet commands:</u>	<u>Press:</u>	<u>Followed by:</u>
BLANK out a block of cells	B	range from:        to:
CALCULATE a block of cells	C	range from:        to:
CALCULATE ACROSS whole matrix	X	-
EXPRESSION	E or SPACE	expression or number
FORMAT a block of cells	F	format (0 to 6) range from:        to:
GO TO selected cell	G	cell reference
LABEL a cell	L	any desired text
NEW spreadsheet required	N	yes (for new model) number of columns number of rows yes (to length)
ORDER of calculation changed	0	
PRINT a block of cells	P	range from:        to:
QUERY formula used in cell	Q	any key (when ready)
REPLICATE	R	replicate from: to:
SAVE	\$ (dollar)	name of file any key (when ready)
SUM a block of cells	S	sum from:        to:
TYPE (on printer)	T	any desired text
UNIVERSAL FORMAT	U	format (0 to 6)
WIDTH of columns changed	W	new width