The Schemaz Tutorial Guide

for RISC OS computers

Norman Lawrence

Edited by Chris Dewhurst

The Schema2 Tutorial Guide

for RISC OS Computers

DRAG 'N DROP
www.dragdrop.co.uk

Contents

Acknowledgements 3

1 . . Getting Started 5

First Spreadsheet 7
Centre Cell Entries 9
Graphs 10
Hotlinking 13
Big Cells 13

2 . . Functions 15

Setting Up the Spreadsheet 15
Processing the Marks 16
Statistics of Results 18
Histograms 19
Sorting the Marks 21

3 . . Importing Files 23

Locking Rows and Columns 25
Multiple Spreadsheets 27
Certificate 28
Workspaces 29

4 . . Matrices 31

Matrix Functions 32
Transposing Matrices 34
Solving Simultaneous Equations 35
Circular References 36
Contour Graph 40

5 . . Macros 43
Minimum Macro. 43
Menu Macros 45
Schema2 BASIC 46
Controlling Repetition 47
Error Conditions 48
Recording a Macro 48
Anatomy of a Recorded Macro 49
BOOT Macros. 51

6 . . Creating graphs. 53
LINEATTRIBUTES 54
LINE 54
BOX 55
ELLIPSE 56
Closed Polygons 57
Path Drawing Functions 58
PARTCIRCLE 59
PARTELLIPSE 60
DRAWOBJECT 61
Open Polygons 63

7 . . A Design Application 67
Resource files 68
PROCAD 70
Listing 7.1 72
Listing 7.2 73

8 . . Interactive Graphics 75
Running the code 75
Dialogue Menu control 76
Parameter String Control 77
Macro Code for Line Graphs 79
Curve Fitting 81
Listing 8.1 84

Appendix 1 Effects, Styles and Formats. 87
Effects 87
Removing Effects 88
Styles 88
Applying a style. 89
Creating a style 89
Editing and Deleting Styles 89
Custom Formats 90
Custom Format Structure 90
Layout of numbers 91
Layout of Dates and Times 91
Sheet Variables 93

Appendix 2 Standard Functions 95
String Functions 96
Algebraic, Trigonometry and Scientific Functions 97
Statistics, Random and Lookup 98
Times Functions 101
Financial Functions. 103

Appendix 3 Standard Macros 105
Column and Row Functions 105
Cell Functions. 107
File Functions. 109
Miscellaneous Functions 110
Access to System SWIs. 112
Bibliography 117
Index 118

Equals icon to use the mouse for entering cell and block names,

Sum icon to place the sequence "=SUM()" into a cell,

Function icon providing access to a wide range of functions.

Below the five icons is the icon ribbon menu, which can be turned on/off from the spreadsheet menu by selecting View and Icon Ribbon or with the short cut key Ctrl+R, see Figure 1.3.

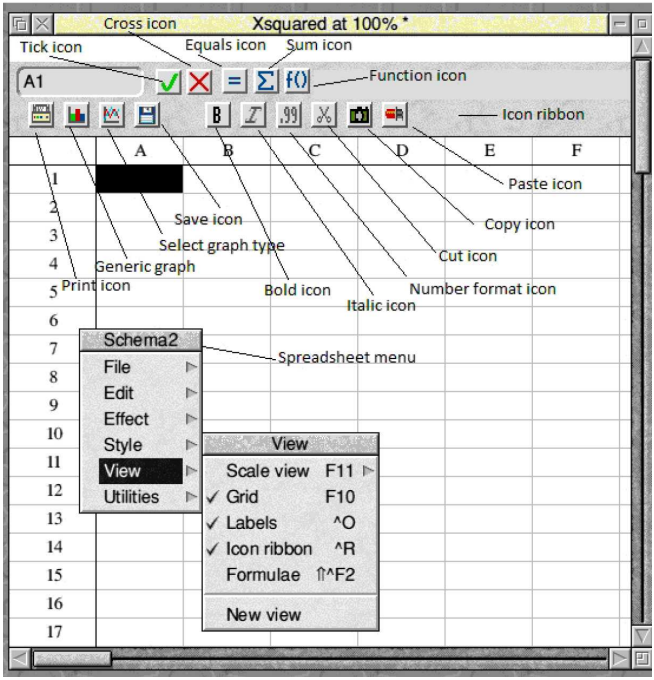
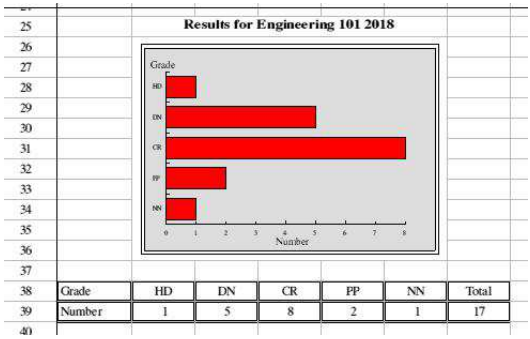


Figure 1.3 Schema2 Spreadsheet icon menus.

First Spreadsheet

It is now time to create a simple spreadsheet. This sheet will calculate the square of the numbers 1 to 10, and display a graph of the results. Referring to Figure 1.4, the steps are as follows:

- Type in two column headings, X and Xsquared into the cells A1 and A2.
- Then enter the numbers 1 to 10 in cells A2 to A11 as shown in Figure 1.4.
- Finally, in cell B2 enter the formulae =$A2^2 followed by return. Note the use of $ sign to indicate an absolute reference to the A column.



Grade	HD	DN	CR	PP	NN	Total
Number	1	5	8	2	1	17

Figure 2.6 Histogram showing grade distributions.

Sorting the Marks

The spreadsheet is almost ready for sending to the board of examiners but the results are required to be arranged in descending order. Schema2, provides options for sorting data by Row or by Column. When sorting by Row, complete rows are shuffled about but each data stays in its column. Likewise sort by Column results in columns being moved about. Data can be sorted in Ascending order or Descending order based upon a specified Key column or row. The student results are required to be sorted based on the the marks awarded in column F and this is how to proceed.

- Select cells A5 to G21 and then bring up the sort dialogue box shown in Figure 2.7 by using Menu > Utilities > Sort (or use the shortcut F6).
- Check the Descending radio buttons for sort by Row, Descending sort and set Key 1 to be F.
- Click on the sort button and the results are now sorted in descending order.

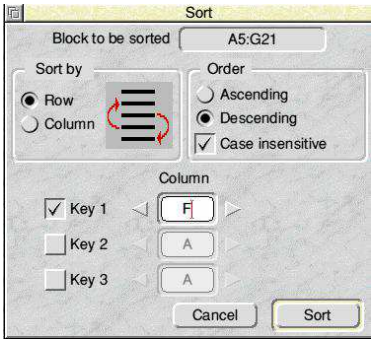


Figure 2.7 Sort Dialogue Box

Solving Simultaneous Equations

Returning to the initial simultaneous equations example containing two unknown values of x_1 and x_2, Schema2 will be used to demonstrate the general principle of solving simutaneous using the MATSOLVE function. Recalling in matrix notation,

$$A = \begin{bmatrix} 2 & 3 \\ 5 & -3 \end{bmatrix} \quad X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad B = \begin{bmatrix} 13 \\ 1 \end{bmatrix}$$

The solution for this expressions is $X = A^{-1} B$ where A^{-1} is the inverse matrix. Most spreadsheets provide a function to find the inverse of matrix A, which can then be multiplied with B to obtain the solution for the column vector X. Schema2 does this in one operation with the MATSOLVE function. Figure 4.6 illustrates the use of MATSOLVE to solve the values for x_1 as 2 and x_2 as 3 which corresponds to the values obtained earlier.

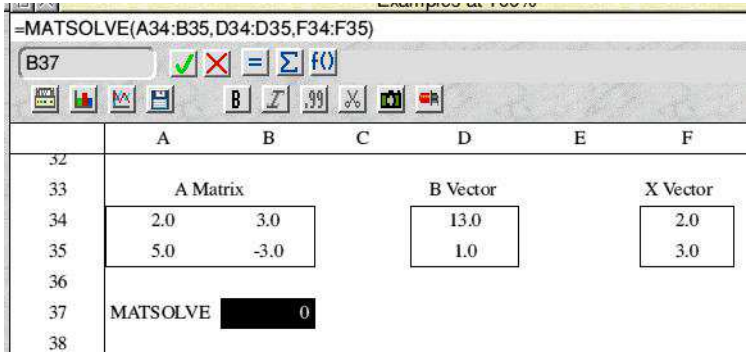


Figure 4.6 Using MATSOLVE

Schema2 works well with simple examples, but how will it cope with more challenging problems? The next example from thermal engineering, requires the solution of 15 simultaneous equations and full details of the problem being solved can be found in the book *Basic Heat Transfer* by Kreith and Black. Suffice to say that the program is required to find the temperature at 15 locations in a 20 cm × 20 cm two dimensional solid with four different edge conditions. Using Schema2 on a Raspberry Pi, the solution time is virtually instanteous, see Figure 4.7 for the Schema 2 sheet, but clearly it would take a considerable time to solve using hand calculations.

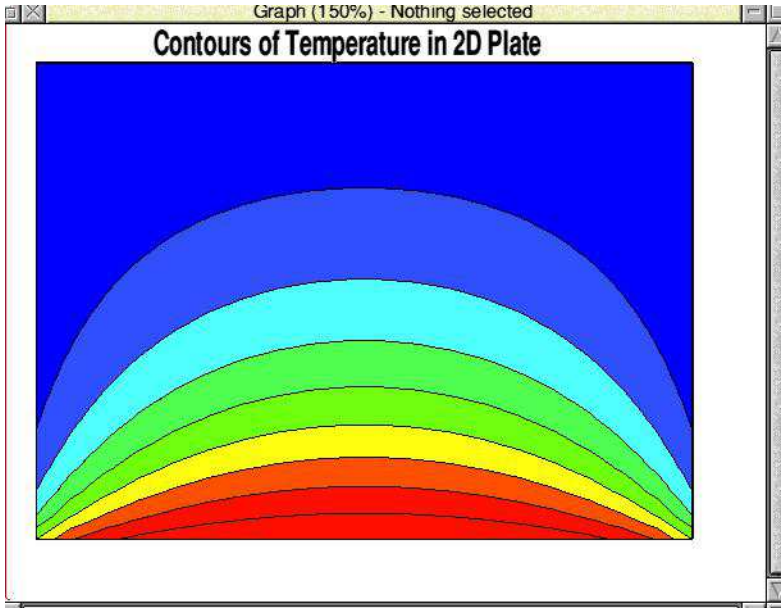


Figure 4.14 Schema 2 Contour graph

It is worth noting that the contoursy graph shown in Figure 4.14 can also be hotlinked to the spreadsheet or saved as a Draw file. The time taken for the hotlinking process is reasonably quick on a Raspberry Pi 4 clocked at 2GHz but slower computers may take some time to complete the task.

```
    endif
  endif
=res
ENDMACRO
```

Controlling Repetition

FOR statements (including nested FOR statements) work as in BBC Basic. The default step is 1 but other steps can be specified, negative or decimal numbers are valid.

Example: This macro will calculate the sum of the numbers between 1 and n

```
MACRO mycount(n)
LOCAL icount,sum
sum = 0
FOR icount = 1 TO n
  sum = sum+icount
NEXT
=sum
ENDMACRO
```

REPEAT. . .UNTIL provides an alternative way of controlling iterations. The terminating condition is made at the end of the loop, so the loop will be executed at least once.

Example: This macro will also calculate the sum of the numbers between 1 and n

```
MACRO mycount(n)
LOCAL sum, icount
sum = 0
icount = 0
REPEAT
  icount = icount+1
  sum = sum + icount
UNTIL icount = n
=sum
ENDMACRO
```

WHILE. . .ENDWHILE also provides an alternative way of controlling iterations. The terminating condition is evaluated at the start of the loop.

Example: In this example, icount is incremented by 1 every time and takes the values 1 2 4 8 etc. until icount = n and the expression in the WHILE statement becomes false and the loop terminates.

```
=w
ENDMACRO
```

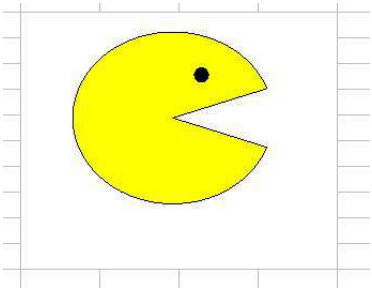


Figure 6.2 Munchie

Closed Polygons

Polygons may be open or closed. Closed polygons are drawn using the POLY function while open polygons use the OPENPOLY function. POLY has the form

res = POLY(w,points,number,colour)

The parameters are:

w	The handle of the graph.
points	A two dimensional array or a block of spreadsheet data.The block must be two rows high and may include any number of columns. Each column gives the x and y co-ordinates of one point in the polygon.
number	The number of points to be plotted from the array. Points beyond this number are ignored.
colour	The fill colour of the inside of the polygon. This is an RGB value and a value of -1 signifies transparency.

The polygon is drawn with the line type specified by the most recent LINEATTRIBUTES function. The first and last points are joined, and the polygon is filled with the specified colour (which may be transparent).

Example: The following macro example uses the POLY function to draw an image of a five sided star (see Figure 6.3) inside an ellipse.

```
MACRO closedpoly
REM Using POLY to draw a closed polygon
LOCAL res,w,j,k,points(1,5),z
```

7 A Design Application

Spreadsheet applications are a great way to help engineering students with their understanding of shell and tube heat exchanger design. For given input conditions, such programs enable students to see important results including length and diameter of the bundle of tubes to be used in the design of a heat exchanger. Of particular help is a graphical display of the orientation of the tubes, to illustrate the series of holes according to the number of tubes and pitch pattern. The effects of pitch patterns such as triangular, square or a square rotated by 90 degrees on the design layout can be studied, see for example, Figure 7.1 where the tubes are arranged in a triangular pitch pattern.

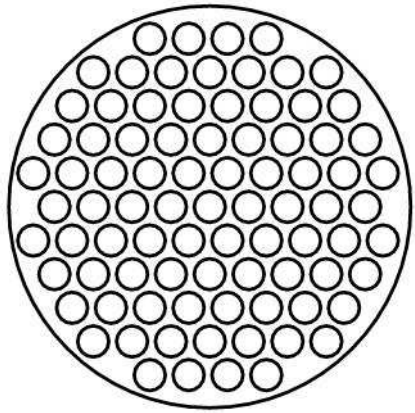
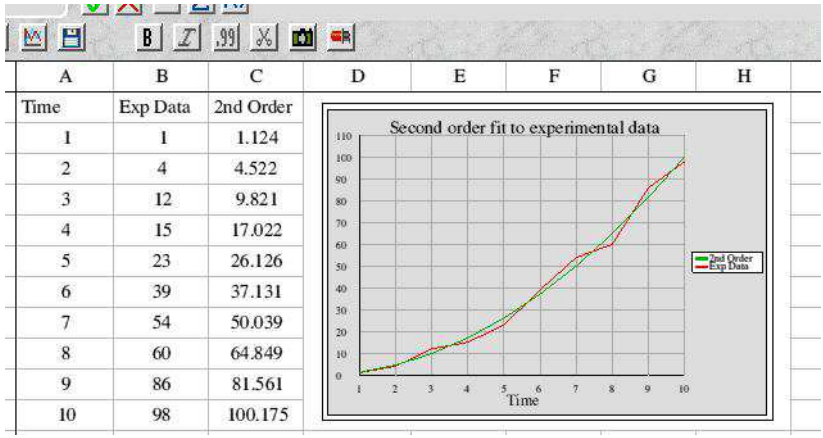


Figure 7.1 Eleven row tube bundle, arranged in a triangular pitch pattern.

This type of application is perfect to demonstrate the graphical capabilities of Schema2, but to keep things simple, the Schema2 macro created for this article is limited to an odd number of rows that are arranged in a triangular pattern. The macro, which is named named circle, is listed at the end of the chapter. Basically, it uses the ELLIPSE function to create a large circle representing the tube bundle diameter and then many small circles according to the number of tubes used. The compiled macro circle was placed into cell F1 in the spreadsheet HeatEx and enlarged to a Big Cell. The sample input data consisted of three values; for the end plate diameter (194), the tube diameter (15) and the horizontal pitch (18.75). This is illustrated in Figure 7.2 together with graphical output of tube bundle arrangement. Calculated data such as the



A	B	C
Time	Exp Data	2nd Order
1	1	1.124
2	4	4.522
3	12	9.821
4	15	17.022
5	23	26.126
6	39	37.131
7	54	50.039
8	60	64.849
9	86	81.561
10	98	100.175

Figure 8.3 Second Order Polynomial Plot

It is easy to extend the graph options to include options for a straight line and cubic curves because they are virtually identical with the exception that constant Porder is 1 for the straight line and 3 is the value for the cubic. Just copy the code typed in for Listing 1 and remember to change the macro name appropriately. Then adjust the line at the end of the macro according to the type of fit required.

```
for j = 1 to h-1
 a(j,w-1) = AB(0)+AB(1)*a(j,0)+AB(2)*a(j,0)^2
next j
```

That is remove the term +AB(2)*a(j,0)^2 for the linear case and add a term +AB(3)*a(j,0)^3 for the cubic case. If all goes well in the typing, copying and pasting then Schema2 should be able to curve fit linear, quadratic and cubic curves to experimental data. If required, the options can be extended to include even higher order polynomials.

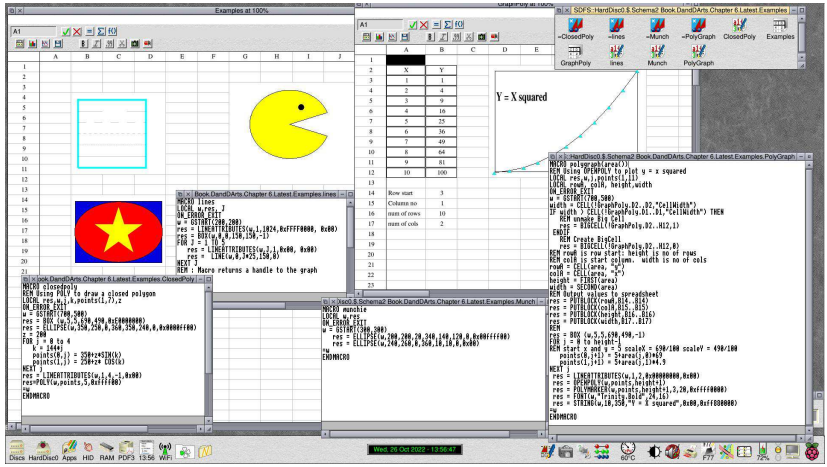
One important requirement is that when selecting the experimental data set, one extra column must be selected for the data fit values to be inserted. The extra column acts as a dummy set of values and it does not matter whether it is empty or contains information, it will be populated by the calculated values.

Some ideas to enhance the best fit capabilities of this macro include, extending the code to include row data input, inserting the coefficients on the spreadsheet and plotting only markers for the experimental data. This will be left as an exercise for the more adventurous readers.

The Schema2 Tutorial Guide for RISC OS computers

Schema2 for RISC OS is a highly extensible spreadsheet application and you can use it to do so much more than the usual point-click-type data manipulation. In this book, Norman Lawrence begins by taking you through the basics – formatting data, formulae and functions including matrix operations and plotting graphs, along with handling Microsoft™ Excel files.

Schema2's real power lies in its macros which are written in a BBC Basic-like language. Numerous examples are provided, starting with drawing interactive and passive graphics. Once you have mastered these concepts you will realise how Schema2 is at home in many fields – business, engineering, commerce and teaching to name a few. An entire chapter is devoted to a design application to help students and the appendices form a concise reference for the styles, effects, functions and macros in Schema2 for RISC OS.



ISBN 978-1-80352-284-5

Schema2 running on a RISC OS Pi demonstrating its ability as both a spreadsheet application and as a powerful macro-driven piece of software.

Front Cover Design: Sybil Harris

:0 Discs :0 16GbPi Apps

DRAG'NDROP
www.dragdrop.co.uk