REVISED NUFFIELD ADVANCED SCIENCE

PHYSICS

DYNAMIC MODELLING SYSTEM

RML AND APPLE HANDBOOK

Author
Jon Ogborn
Department of Science Education
Institute of Education
University of London

Programmers
David Creasy
Chelsea College
University of London

Dennis Wong
Department of Science Education
Institute of Education
University of London

Jon Ogborn

o Software

Preface

This computer assisted learning unit has been produced by Professor Jon Ogborn supported by the Computers in the Curriculum Project at Chelsea College. It was designed to be used in the Revised Nuffield A-Level Physics course as well as other secondary science courses.

The program has two parts: the operating system and a Physics models disk. The program is supported by a comprehensive guide on how to use the system with examples of how to create a model. A guide to the models disk is also provided, which describes each of the Physics models on the disk. The user is also able to create and save new models, thus building up a library of models to meet individual needs.

The system has been designed to meet the needs of different teaching methods, student abilities and class groupings. Additional guides and models disks are being developed for other subjects, for use in further and higher education as well as in the secondary school curriculum. This system has a wide range of applications in science teaching and can also be used in many other areas of the curriculum.

The Dynamic Modelling System is part of a wide range of computer assisted learning materials developed by the Computers in the Curriculum Project for use in education.

Margaret Cox
Project Director
1985

Copy 7

Contents

Acknowledgements

This system has been developed in conjunction with the Computers in the Curriculum Project, part of the Educational Computing Section, Chelsea College. The principal author is Professor Jon Ogborn (Institute of Education).

The development of the system has been partly funded by the Nuffield Chelsea Curriculum trust.

The work has been directed by David Squires (Project Science Manager). Valuable assistance has been provided by Sophie McCormick (Assistant Director Science), David Creasy and Dennis Wong (Institute of Education).

Thanks are also due to the Central Programming team for their help and advice in the development of this program and the secretarial team in typing the guides.

Margaret Cox
Project Director
1985

1 Introduction to Dynamic Modelling

The Dynamic Modelling System is a general purpose tool for use by teachers and students. It will plot graphs or tabulate values of any quantities the user instructs it how to calculate. This is done either by writing a fresh set of instructions, usually only a few lines long or by using a previously prepared set of instructions stored on disk.

The system can be used in a range of subjects and in a variety of ways. These include:

- demonstration of a model;
- building up a model by discussion;
- exploring a model in group or individual work;
- developing a new model, for example for project work;
- teaching elementary programming.

A dynamic model of a real situation is a sequence of steps in which the way the system will evolve is calculated. Successive stages of its evolution are found by repeating the sequence, starting with the values found at the previous stage.

A very simple model of population growth will illustrate the idea of a dynamic model. For a population of rabbits, the new number of rabbits in each generation is the old number of rabbits plus the number of births. This can be expressed as below:

RABBITS = RABBITS + BIRTHS
or R = R + B

Suppose we start with two rabbits (R = 2) and add four births (B = 4) in each generation. This model would then generate the values R = 2, R = 6, R = 10, R = 14, and so on. All models in the Dynamic Modelling System work inside a repeating loop which calculates once through the instructions given, and then does this again and again, each time using the values from the last time around. Adding a second instruction to add one to the number of generations (say G) at each step can be done by inserting the line G = G + 1:

R = R + B
G = G + 1

A graph of R against G would be a straight line. Suppose we alter the model, and say that the number of births is proportional to the number of rabbits there happen to be:

B = F*R
R = R + B
G = G + 1

F might stand for fertility. Now the growth becomes exponential, the graph of rabbits against number of generations rising faster and faster. Next it would be possible to introduce a mortality rate, and perhaps to make that and the fertility rate depend on the density of the rabbit population or on other factors.

Models for the decay of charge on a capacitor or for the decay of radioactive materials are nearly as simple. Something similar to the rabbit model could represent an explosive chemical reaction.

The point of the example is that:

- models can be built up from very simple starting points;
- being written as elementary steps for the computer, each step is usually easy to understand, and normally has a direct interpretation in terms of events in the situation being modelled;
- difficult mathematical functions (such as the exponential) need not be introduced before one can get results, and models which have no analytic solutions, or ones which are too advanced, can still be examined.

Examples of models

The Dynamic Modelling System can be used in a variety of subjects. A small selection of examples is given below. More extensive ideas can be found in the notes supplied with the models disk.

All the models are written using the syntax found in the programming language BASIC, for example the * symbol is used to represent multiplication.

Biology

The model below gives a logistic growth curve of population P against time T. G is a net growth rate, which diminishes from the base rate GO as the population approaches a maximum M:

$$G = GO*(1 - P/M)$$
$$P = P + G*P$$
$$T = T + 1$$

Physics

Free fall under gravity can be modelled by

$$F = -G*M$$
$$A = F/M$$
$$V = V + A*DT$$
$$S = S + V*DT$$
$$T = T + DT$$

The force F, determined by the gravitational field G and mass M, is used to calculate the acceleration A. The new velocity after time DT is found by using A, and the change of displacement after DT is found by using V.

The model above is an example of how the system can be used to solve a differential equation. Another example would be that of capacitor discharge:

$$Q = C*V$$
$$I = V/R$$
$$Q = Q - I*DT$$
$$V = Q/C$$
$$T = T + DT$$

Q is the charge on a capacitance C at potential difference V, discharging with current I through resistance R.

Chemistry

A first order reaction would be very much like the last example:

$$DC = -K*C*DT$$
$$C = C + DC$$
$$T = T + DT$$

C is the concentration and K is a rate constant.

Economics, Geography

Resource depletion is one example amongst many. An introductory model might be:

$$G = K*(R - CL)/CL$$
$$C = P*R + G*C$$
$$R = R - C$$
$$T = T + 1$$

Resources R are consumed at a rate C, which is a proportion

P of present resources plus growth G in consumption. G is decided by the present resources in relation to a critical level CL.

Mathematics and Computing

At an elementary level functions can be plotted. The behaviour of the sine function can be shown by:

```
S = SIN(W*T)
T = T + DT
```

2 Using the system

Outline of the system

You can see a short demonstration of the program and its commands, by pressing I (Information) when the introductory menu appears at the start of the program. Pressing M (Modelling) takes you into the modelling program itself.

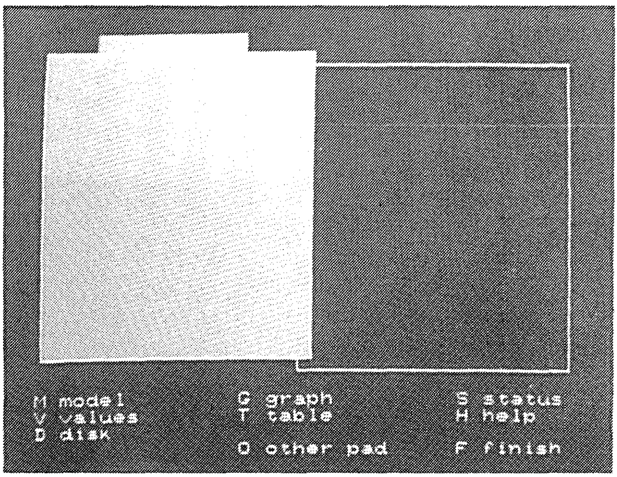


Figure 1 The screen at the start of the modelling program

Figure 1 shows the screen at the start of the modelling program. On the screen, the program gives you two working pads, side by side. One pad can be used to display information you need for the work you are doing on the other pad or you can compare results from two versions of the same model by using separate pads. Only one pad can be described at a time. On either pad you can do any one of six kinds of task described overleaf.

Model
This allows you to type in a model, read one from the models disk, alter a model you have or just look at it. The system provides a simple screen editor for typing in models, modifying them or correcting errors.

Values
This is used to type in, view or modify initial values for variables in a model. The editor used for models is also available here.

Graph
Graphs may be plotted using any two variables in the model. The positions of the axes and scales can be varied.

Table
This produces tabulated values of any pair of variables in the model.

Status
This checks whether the model, values or the choice of axes and scales, are yet to be specified.

Disk
This lets you see the names of models stored on the disk, view their content, and save a new model or delete an old one.

3 An exercise in using the system

In the exercise below you will see most of the system facilities at work. It begins on one pad. You type in a model on that pad using the editing facilities, and then, on the other pad, you give some initial values to the variables, again using the editor. On this pad, graph axes are set up and a graph is plotted. The model is still visible on the first one.

Next, the values replace the model on the pad opposite the graph, a value is changed and a new graph is plotted.

Then the model itself is altered by bringing that back on a pad and editing it. The effect of the change is shown by having the new graph drawn side-by-side with the old one, using both pads.

Finally you see how to store a model on the models disk, and how to recall one from the disk.

Start up the system with the Dynamic Modelling System disk in the disk drive (drive A or drive 1 of a double disk drive – see the 'System description' section for technical details).

The introductory menu appears on the screen:

M	Modelling
I	Information
S	Settings (with current settings shown)
Q	Quit

- Press **M**

The two pads appear on the screen. Both are empty, and you begin working on the lefthand pad. The menu shows the commands for the tasks you can perform on the pads. This is shown in Figure 1.

Writing a model

- Press **M**

This labels the lefthand pad MODEL and brings up a menu of commands for writing models.

You can now either type model equations onto the pad or use the command keys, for example to move the cursor.

The editing keys are letters pressed with the CONTROL (CTRL) key held down. Thus CTRL U moves the cursor up. The menu shows CTRL U as ↑U. The CTRL key is used so as to distinguish command keys from characters being typed into a model. On the 480Z microcomputer the arrow keys also move the cursor.

First we will type in a model.

- Type **R = R + B (RETURN)**

This equation is printed on the pad and is now part of the model.

According to the equation, the new value of R is to be assigned the old value plus B. All assignments in BASIC may have only one variable on the left of the = sign.

If you make typing errors, use the DELETE key or press CTRL W to rub out a letter, or CTRL K to rub out the whole line. CTRL N makes space for a new line between two others. Such commands take effect at the current cursor position.

- Type **G = G + 1 (RETURN)**

This line is now added to the model.

- Type **B = F*R (RETURN)**

This line is added. Now there is a problem: this line should be the first, not the last, since B is needed to calculate R. However, the editor allows lines to be moved up or down.

- Use **CTRL U** or **CTRL D (arrow keys on 480Z)**

Position the cursor on the line which reads B = F*R.

- Press **CTRL T**

This moves the last equation up one line. Repeat until it becomes the first equation.

You should now have the following equations on the pad, as shown in Figure 2:

B = F*R
R = R + B
G = G + 1

Setting values

- Press **ESC**

When the ESCape key is pressed, the current task is terminated and the main menu showing pad uses is displayed.

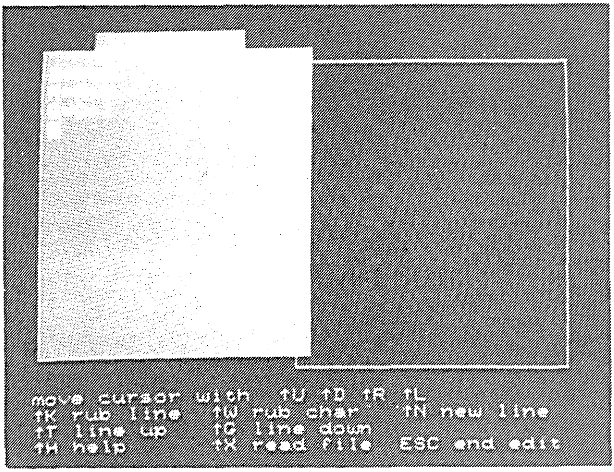


Figure 2 Entering a model

- Press **O** (letter O not zero)

This will move you to the other pad.

- Press **V**

This enables you to use the new pad for initial values.

- Type **R = 2 (RETURN)**
- Type **G = 0 (RETURN)**
- Type **F = 0.1 (RETURN)**

These will start the model calculation off at generation zero with two rabbits. The fertility is 0.1.

The values can be edited using the same CTRL keys as were used in the model option.

Plotting a graph

- Press **ESC**

Now you can choose another use for the pad without changing the pads.

- Press **S**

This gives the model a status check, to see if anything has not been set. You have written a model and its values, but not yet chosen the graph axes or scales. So you should get the message:

No *x*-variable
No *y*-variable

No x-maximum
No y-maximum

- Press **G**

The righthand pad will show the graph axes.

- Press **Y**

This asks for the variable to go on the Y-axis.

- Type **R (RETURN)**

This will place R on the Y-axis.

- Press **X**

This asks for the variable to go on the X-axis.

- Type **G (RETURN)**

This will place G on the X-axis.

- Press **M**

This asks for the largest value of R.

- Type **200 (RETURN)**

The maximum value of R is set at 200.

- Press **N**

This asks for the largest value of G.

- Type **50 (RETURN)**

The maximum value of G is set at 50.

- Press the **U** and/or **R** keys **(arrow keys on 480Z)**

This moves the axes. They can be put back in place by pressing D and/or L. The zero on the graph is assumed to be at the origin of the axes, so that the scales are decided by giving the maximum values. The origin may be put anywhere on the pad.

The system is now ready to do calculations and plot the results. The preliminaries take less time as you get used to the system, and since models complete with all other data can be stored, such a stored model can run almost immediately after starting.

- Press **B**

The system checks the equations and values before starting to plot. Messages appear as this happens. With the values above you should get a good exponential curve.

- Press **W**

This clears the graph pad.

- Press **B**

This runs the model again.

- Press **S**

This stops the graph while it is being plotted.

- Press **C**

This continues the graph plotting.

We are now going to call up the initial values again, change the fertility rate and plot a new graph on top of the previous one.

Editing

- Press **ESCape,** then **O**

- Press **ESCape,** then **V**

This stops work on the graph, and the program changes to the lefthand pad and starts to use it for initial values. Figure 3 shows the graph on one pad and values on the other.

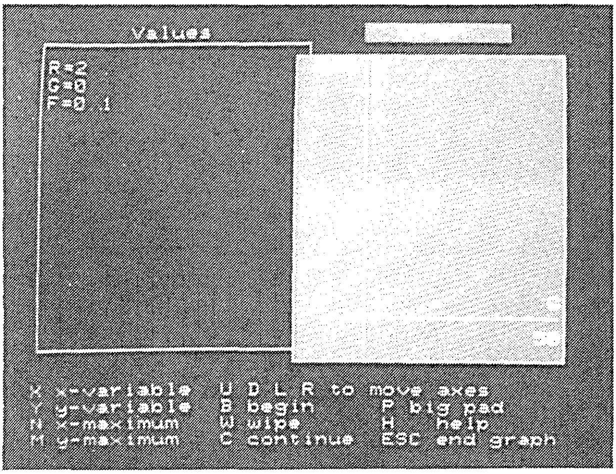


Figure 3 Looking at values and a graph

To move from one pad use to another, press ESCape followed by one of the pad use commands.

- Use **CTRL** and the **U, D, R** and **L** keys

Put the cursor on the '1' in the line F = 0.1.

- Use **CTRL W**

Delete the '1' and then type '2', making F = 0.2.

If you omit to use .CTRL, the program 'thinks' you are writing in new variables. Rub out any mistakes with CTRL W or DELETE.

- Press **ESC, then O**

This returns you to the righthand pad which takes up its previous use as a graph plotter.

- Press **B**

The exponential growth becomes more rapid than previously.

Having altered a variable, we will next alter the model itself, making the fertility dependent on how crowded the rabbits are, and therefore not constant.

- Press **ESC, then O**

This takes you to the other pad.

- Press **ESC, then M**

This enables you to use models. The model equations appear. Move the cursor to the top line, and press CTRL N to make a space for a new line. Type the new first equation:

F = FO*(1 – R/M)

This makes F decrease as the number of rabbits gets closer to a maximum value M.

We now have to specify the initial values of the base fertility FO and the maximum M.

- Press **ESC, then V**

This puts values on the same pad.

- Use **CTRL** and the **U, D, R** and **L** keys **(arrows on the 480Z)**

Put the cursor after F in F = 0.2 and type 'O' to make it read FO = 0.2. Move the cursor to the end of the variables and type M = 100. Now the two new variables which have been introduced have initial values.

- Press **ESC, then G**

This sets up the current (lefthand) pad for a graph to be plotted, leaving the original graph on the righthand pad.

- Press **B**

You show now obtain a growth curve which rises at first and then reaches a maximum. With the two graph pads side-by-side the graphs can be compared, as shown in Figure 4.

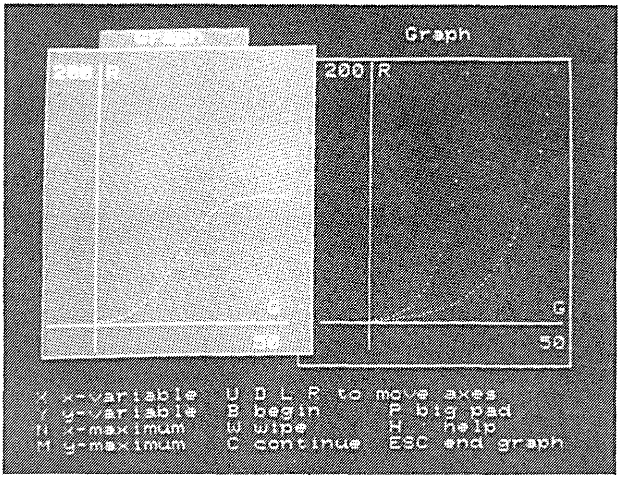


Figure 4 Comparing graphs

Different types of output

- Press **P**

This gives, on the big graph pad, a graph suitable for demonstration.

- Press **B**

The program will start plotting the graph.

- Press **P**

This returns the program to the two pad system.

- Press **ESC, then T**

This sets up a table of values to be printed on the pad. The variables tabulated are those chosen for the graph, but can be altered as for the graph by using the X and Y keys.

- Press **B**

This begins to tabulate values.

- Press **C**

More tabulated results are provided.

It is often very useful to see a table before plotting a graph so as to get an idea of appropriate scale values. A table will often show why a graph appears not to plot, for example if the values are off the scales or are all zero.

Saving a model

- Press **ESC,** then **D**

This shows the names of any models at present stored on the models disk.

- Press **S**

This asks for the model you have just written to be saved (stored) on the models disk. The program asks for a name for the model: use a name which is **not** already in use. Its name will be added to the list of stored models shown on the screen.

- Press **ESC,** then **F**

This asks if you want to finish (F again), to continue (C) with the present model (in case of mistakes), or to start again (S) with an empty system as at the start.

- Press **S**

- Press **M**

This enables you to model.

- Use **CTRL X**

You are asked for the name of the model you want. Type in the name you gave it. A copy of the model is now made from the disk.

- Press **ESC,** then **V**

The initial values are now shown.

- Press **ESC,** then **G**

The graph axes are shown with scales and labels of what to plot.

- Press **B**

This will plot the graph.

- Press **ESC,** then **F**

- Press **F**

This enables you to exit from the program.

4 System description

Machine requirements

You require a computer with a single or double disk drive. Colour and monochrome monitors or television sets can all be used.

Minimum specifications:

RML380Z	56K
	High Resolution Graphics board
	COS 3.4 or later
	Any BASIC supporting high resolution graphics
RML480Z	64K
	Any BASIC supporting high resolution graphics
Apple II	
Apple IIe	48K
	DOS3.3 and Applesoft Basic in ROM or
	DOS3.2 and BASICS disk
Apple IIc	

Starting and setting up the system

You need **two** disks: the Dynamic Modelling System disk, and a second models disk which you will use to call up or to store models. If you are not using the models disk supplied, you need a formatted disk for this purpose.

On a two disk system, the Dynamic Modelling System disk goes in drive A (drive 1 in an Apple microcomputer), and the model disk in drive B (drive 2 in an Apple microcomputer).

If you have a single disk drive, insert the Dynamic Modelling System disk in the drive first. When required you will be prompted to insert the models disk. The HELP commands require the Dynamic Modelling System disk to be replaced.

The system is supplied set for a **single** drive and a **monochrome television/monitor**. This may be changed by pressing S when the initial menu is displayed. A double disk drive configuration of the software and/or a colour display can then be selected.

System facilities and restrictions

Use of BASIC in models

Models are written as lines of BASIC. They will be computed in sequence, looping to the beginning after plotting or tabulating values from the current iteration. Thus the sequence of instructions is usually important.

Line numbers are not used; the program inserts them when it uses the model. Each line on the screen is treated as a new instruction. Thus lines **cannot** carry over from one line to the next.

The system does **not** check if the model is 'correct', and only makes a limited check on syntax. If anything is wrong, and the computation fails, you will get an error message.

The system replaces your variables with its own ones, so you can use any letters you wish for variable names. Lower case variable names can be used on all machines except the Apple II. When the machine supports lower case, a lower case and upper case variable of the same name will represent different variables.

In writing models and variable values you can use the arithmetic operators:

+ (addition)	− (subtraction)
* (multiplication)	/ (division)
= (equals)	↑ (exponent)

You need not use LET statements, though you may do so. You can use the colon (:) for multiple statements on one line.

The system also allows the use of the following BASIC functions:

SIN-sine
COS-cosine
TAN-tangent
ATN-arctan
LOG-logarithm
EXP-exponential
SQR-square root
INT-next lower integer value
ABS-absolute positive value
RND-random number in range 0 to 1

These functions require brackets, for example LOG(R). Brackets may also be used in arithmetic expressions, for example $X = Y*(P + Q)$. All angles should be expressed in radians. The random number generator should be written as RND(1).

The functions, or arithmetic expressions, may also be used either in giving initial values, or in labelling axes or tables. Thus, one can plot LOG(R) instead of R, or set an initial variable equal to a random number.

The system does **not** recognise any other BASIC keywords, and if they are used, it will treat them as if they were names of variables.

The following logical control functions can be used:

= < >

IF THEN

You cannot use array variables in the restricted BASIC available to the system.

Size of a model

The number of lines in a model, and the number of characters on any one line, is restricted to what will fit on a pad. Lines which are too long to fit will need to be rewritten as two lines. For example, the energy E below:

E = 0.5*M*(VX↑2 + VY↑2) + M*G*SY

written as two terms, could be split into its two parts, adding them afterwards:

KE = 0.5*M*(VX↑2 + VY↑2)
PE = M*G*SY
E = KE + PE

Screen editor

A screen editor is used for writing or modifying models or initial values of variables. It provides for:

- typing characters to the screen;
- deleting characters at the cursor or to the left of the cursor;
- erasing a whole line;
- inserting an empty line;
- moving a whole line up or down.

Models can be read in from disk, when in the editor. A model read in is appended to any model that currently exists. It may be necessary to move all or part of it, using the line up or line down facilities, to get the correct sequence.

Disk storage of models

Models are stored on (saved to) and read from a models disk. This may be one supplied with the system, or a blank disk (formatted for the machine in use) supplied by the user. In this way, different teachers or students can have their own models disks while using the same modelling system.

The models are stored with the names you give them. Names must not include punctuation marks or be more than seven characters long. They must also start with a letter.

Your models disk will have put on it a file called CAT.DMS, which contains a catalogue of your models on disk. This file must not be erased.

Graphs and tables

The origin is defined as the point where the axes of the graph cross. They can be moved to place the origin where it is needed. Scales are chosen by giving the maximum positive values of each variable. It follows that the scaling changes if the axes are moved and the maxima are left the same.

The tabulating facility gives numerical values of a pair of variables. The two variables are in common with the graph facility so that what is plotted is what is tabulated, and vice-versa. It is of course possible to alter what is tabulated or plotted. Values tabulated are shown with three significant figures.

Help messages

Help messages are available to give a fuller explanation of each menu of commands, obtained with key H, or, in the editor, CTRL H (CTRL Z on the Apple microcomputers).

In graph mode, the system tries to detect if the plotted point went off the screen. In this case the help message suggests using a table to inspect values.

Crashing

The Dynamic Modelling System is a program that, when instructed by the user, programs itself. Thus, it can crash as can any program you write directly, for example, if you divide by zero. If this happens, the error is 'trapped' and the program allows you to start again or corrrect your fault.

Replies to unwanted prompts

If a command key such as that needed to save a model or to label an axis is accidently pressed, the system will be waiting for a reply, such as the name of model, or a variable to plot. To cancel the command press the RETURN key.

5 Guide to system commands

A chart showing the menus of commands in the system, and how each is reached from others, can be found on the back cover of this booklet. The central PAD USES menu gives access to the other menus, and leaving the others by pressing the ESC key returns to the PAD USES menu.
The descriptions which follow give fuller information about the use and effect of the various commands.

Introductory menu

M start up the modelling program.
I give an introductory demonstration which returns to the introductory menu.
S allow current settings (displayed) to be altered.
Q quit from the modelling system.

Pad uses menu

Entered at the start of modelling, and returned to from the other menus by pressing ESC. The use selected is set up on the currently active pad.
M write, edit, or read in a model from disk.
V write or edit the initial values of variables.
D save, delete or view models stored on disk.
G set up the graph axes for plotting.
T set up the table for tabulating numerical values.
S check the status of the model to detect the absence of model, values, or graph specifications. Called up automatically if an attempt is made to compute with any of these missing.
H help information on the current menu (available in all menus).
O switch to the other pad not currently active, to pick up its previous use, and make it the active pad.
F finish exit from the system.

Editing menu

Used by both the MODEL and the VALUES uses of a pad. Commands in this menu use CTRL keys (shown by ↑ on the screen). Characters typed without CTRL appear on the pad as part of a model or a set of values.

↑U↑D↑R↑L	move the typing cursor.
DELETE	rub out the character to the left of the cursor.
↑W	rub out the character at the cursor.
↑K	rub out the whole line at the cursor.
↑N	insert an empty line at the cursor.
↑T	move the line at the cursor one line up.
↑G	move the line at the cursor one line down.
↑X	read a file from disk, asking for its name. The file is appended to any model already present.
ESC	call up the pad uses menu.
↑H	print help with the editing menu for RML microcomputers.
↑Z	print help with the editing menu for Apple microcomputers.

Graph menu

U D R L	move the graph axes.
X	choose or alter the *X*-variable plotted.
Y	choose or alter the *Y*-variable plotted.
N	choose or alter the maximum of the *X*-variable.
M	choose or alter the maximum of the *Y*-variable.
B	begin plotting: stops at the edge of the pad.
C	continue plotting (after a halt).
W	wipe the pad and redraw the graph axes.
P	switch between the large or small graph pads.
ESC	calls up the pad uses menu.
H	prints help with the graph menu.

Table menu

X	choose or alter the first variable tabulated.
Y	choose or alter the second variable tabulated.
B	begin tabulating: stop when the pad is full.
C	continue tabulating (fill the pad again).

ESC calls up the pad uses menu.
H prints help with the table menu.

Disk menu

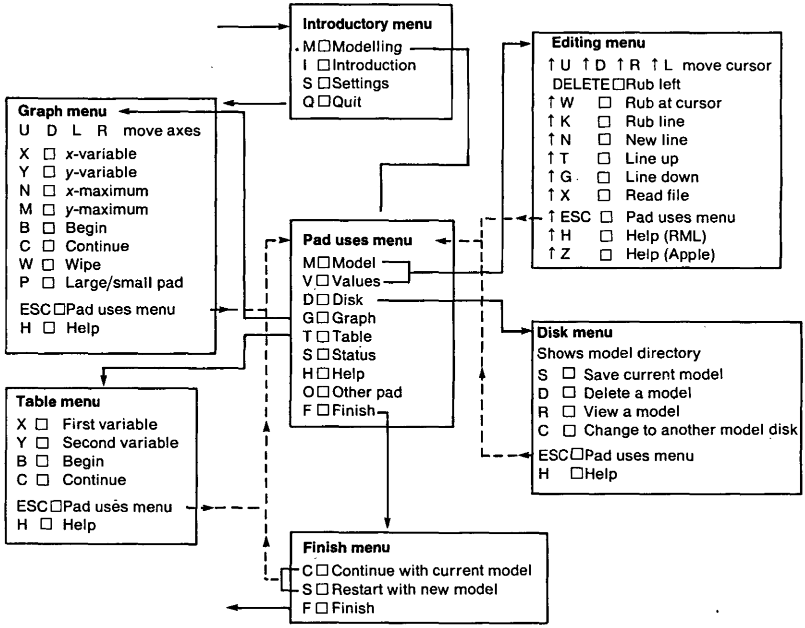
When D is pressed, the names of models filed on the disk are displayed.

S save the current model on the disk. A name is requested.
D delete a named model from the disk.
R view a model on the disk. This model is **not** made part of the current model.
C change to another models disk.
ESC call up the pad uses menu.
H prints help with the disk menu.

Finish menu

This is reached by pressing F in the pad uses menu. It is also reached if an error occurs; an error message is printed.

C continue with current model.
S start again with no model in the system.
F finish finally with the modelling system.


Introductory menu
M □ Modelling
I □ Introduction
S □ Settings
Q □ Quit
Editing menu
↑U ↑D ↑R ↑L move cursor
DELETE □ Rub left
↑W □ Rub at cursor
↑K □ Rub line
↑N □ New line
↑T □ Line up
↑G □ Line down
↑X □ Read file
↑ESC □ Pad uses menu
↑H □ Help (RML)
↑Z □ Help (Apple)
Graph menu
U D L R move axes
X □ x-variable
Y □ y-variable
N □ x-maximum
M □ y-maximum
B □ Begin
C □ Continue
W □ Wipe
P □ Large/small pad
ESC □ Pad uses menu
H □ Help
Pad uses menu
M □ Model
V □ Values
D □ Disk
G □ Graph
T □ Table
S □ Status
H □ Help
O □ Other pad
F □ Finish
Disk menu
Shows model directory
S □ Save current model
D □ Delete a model
R □ View a model
C □ Change to another model disk
ESC □ Pad uses menu
H □ Help
Table menu
X □ First variable
Y □ Second variable
B □ Begin
C □ Continue
ESC □ Pad uses menu
H □ Help
Finish menu
C □ Continue with current model
S □ Restart with new model
F □ Finish