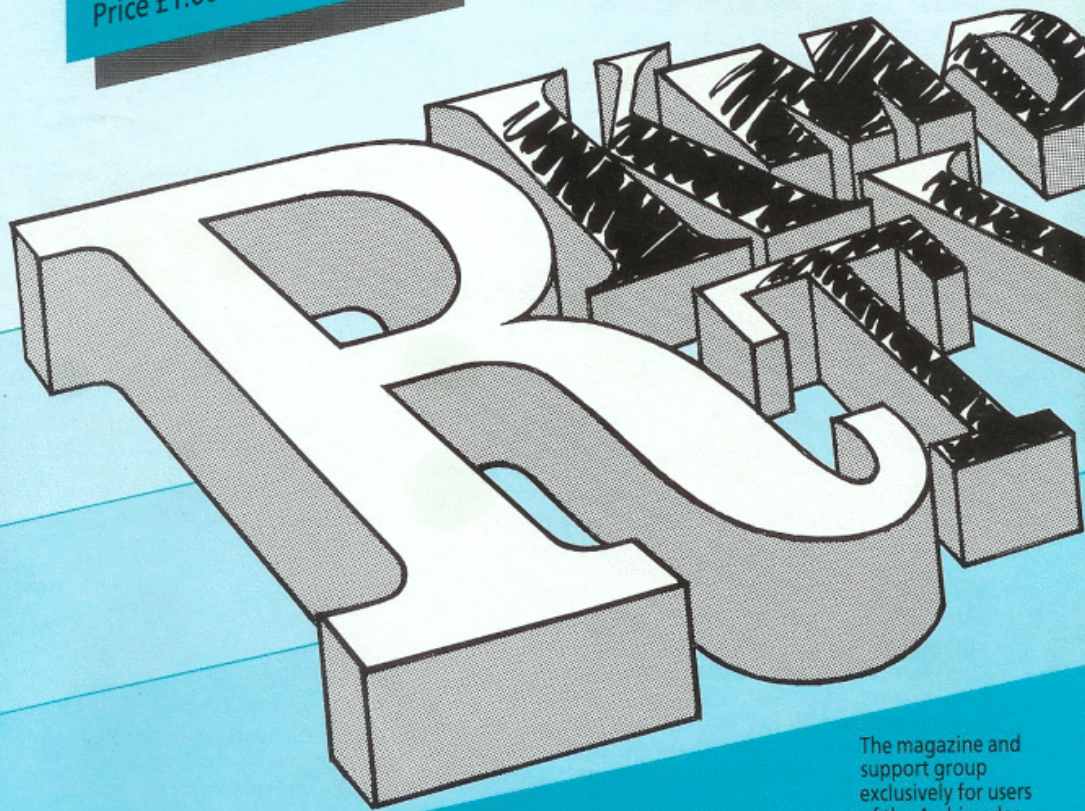Filetype Setter

System Fonts

SCSI Interfaces

Sound Module Generator

Outline Fonts

The magazine and
support group
exclusively for users
of the Archimedes

A Check Save Module

Peter Arnold describes a short utility which can prove invaluable in preventing the accidental overwriting of files.

The short module listed here provides a useful function, particularly for program developers, but others as well. Whenever a *save* function is to be performed, the utility, once installed, checks to see if any file with the same name (and pathname) already exists, and if so seeks confirmation from the user before proceeding. This can be invaluable for preventing the inadvertent overwriting of files, whether saved directly (from within Basic or from the command line, for example), or from the Desktop (by dragging). In the latter case, pressing the Select button confirms ('Y'), while any other button cancels.

The utility is supplied in the form of a module called *RMCheck*, which can be assembled by entering and running the program listed here (but do save the source code before running it). It would also be advisable to save a copy of the original program once entered (use the name *CheckSave*) in case any mistakes have occurred. Once the module has been created, it can be installed by double clicking on its icon in a directory viewer, or by typing:

    *RMLOAD RMCheck

in response to the system prompt (*).

Once loaded, the module is immediately active. However, two star commands have also been provided, *CheckOff* and *CheckOn* which may be used to switch the check off, or back on again. The module will also respond to the use of the *HELP command followed by the module name, e.g.:

    *HELP RMCheck

This module is particularly useful to anyone developing programs in Basic, but also in any situation where it is otherwise

possible to overwrite a file without realising what has happened until maybe too late.

For those who may be interested in the workings of the program, an annotated version is included on the magazine disc.

```
  10 REM            >CheckSave
  20 REM Program    Save Checker
  30 REM Version    A 1.0
  40 REM Author     Peter Arnold
  50 REM RISC User  Jan/Feb 1990
  60 REM Program    Subject to Copyright
  70 :
  80 DIM code% 1000
  90 PROCassemble
 100 OSCLI"SAVE RMCheck "+STR$~code%+"+
"+STR$~P%
 110 OSCLI"SETTYPE RMCheck FFA"
 120 END
 130 :
 140 DEF PROCassemble
 150 FOR pass%=4 TO 7 STEP 3
 160 P%=0:O%=code%
 170 [OPT pass%
 180 .Start
 190 EQUD 0:EQUD Init
 200 EQUD ReleaseVector
 210 EQUD Service:EQUD Title
 220 EQUD Help:EQUD Comms
 230 .Title
 240 EQUS "CheckSave"
 250 EQUB 0:ALIGN
 260 .Help
 270 EQUS "Save Checker"+CHR$9+"1.00 ("
+MID$(TIME$,5,11)+")"
 280 EQUB 0:ALIGN
 290 .Comms
 300 EQUS "CheckOn":EQUB 0
 310 ALIGN:EQUD Checkon:EQUD 0
 320 EQUD Onsyn:EQUD Onhlp
 330 EQUS "CheckOff":EQUB 0
 340 ALIGN:EQUD Checkoff:EQUD 0
```

```
350 EQUD Offsyn:EQUD Offhlp
360 EQUB 0
370 .Onhlp
380 EQUS "CheckOn turns on save checki
ng"
390 EQUB 13
400 .Onsyn
410 EQUS "Syntax: *CheckOn"
420 EQUB 0
430 .Offhlp
440 EQUS "CheckOff turns off save chec
king"
450 EQUB 13
460 .Offsyn
470 EQUS "Syntax: *CheckOff"
480 EQUB 0
490 .Checkon
500 LDR R12,[R12]:MOV R0,#1
510 STR R0,[R12]:MOV PC,R14
520 .Checkoff
530 LDR R12,[R12]:MOV R0,#0
540 STR R0,[R12]:MOV PC,R14
550 .Service
560 STMFD R13!,{R14}
570 CMP R1,#&27
580 BLEQ ClaimVector
590 LDMFD R13!,{PC}
600 .Init
610 STMFD R13!,{R14}
620 MOV R0,#6:MOV R3,#1
630 SWI "OS_Module"
640 STR R3,[R2]:STR R2,[R12]
650 BL ClaimVector
660 LDMFD R13!,{PC}
670 .ClaimVector
680 STMFD R13!,{R0-R3,R14}
690 MOV R0,#8:ADR R1,CodeStart
700 LDR R2,[R12]:SWI "OS_Claim"
710 LDMFD R13!,{R0-R3,PC}
720 .ReleaseVector
730 STMFD R13!,{R14}
740 MOV R0,#8:ADR R1,CodeStart
750 LDR R2,[R12]:SWI "OS_Release"
760 LDMFD R13!,{PC}
770 .CodeStart
780 STMFD R13!,{R0-R12,R14}
790 LDR R6,[R12]:CMP R6,#1
800 BNE Exit
810 CMP R0,#0:CMPNE R0,#10
820 BNE Exit
830 MOV R0,#5
840 SWI "XOS_File"
850 CMP R0,#0:BEQ Exit
860 MOV R9,R1:SWI "OS_WriteS"
870 EQUS "File exists - Overwrite (Y/N
) ?"
880 EQUB 7:EQUB 0:ALIGN
890 BL Entry
900 .Access
910 AND R1,R5,#8:AND R2,R5,#2
920 ORR R1,R1,R2:CMP R1,#2
930 BEQ Exit
940 SWI "OS_WriteS"
950 EQUS "Insufficient access - Contin
ue? (Y/N)"
960 EQUB 7:EQUB 0:ALIGN
970 BL Entry
980 MOV R1,R9:MOV R0,#4
990 MOV R5,#51:SWI "OS_File"
1000 .Exit
1010 LDMFD R13!,{R0-R12,PC}^
1020 .Nosave
1030 SWI "OS_WriteS"
1040 EQUS "Save abandoned"
1050 EQUB 0:ALIGN
1060 SWI "OS_NewLine"
1070 LDMFD R13!,{R0-R12,R14}
1080 LDMFD R13!,{PC}
1090 .Entry
1100 MOV R7,R14:SWI "OS_Confirm"
1110 SWI "OS_WriteC":SWI &10D:SWI &10A
1120 BCS Nosave:BNE Nosave:MOV PC,R7
1130 ]:NEXT
1140 ENDPROC
```

RU