

Volume 2
Issue 4

March
1989

Price £1.20

RISC USER



**RISC
USER**



Image Wrap and Spin

THE MAGAZINE AND SUPPORT GROUP
EXCLUSIVELY FOR USERS OF THE ARCHIMEDES

NEW FONT STYLES

This short relocatable module by Peter Arnold provides 72 new styles for the Arc's normal screen font.

The Font Masher relocatable module is designed to bridge the gap between the Archimedes' normal screen font (BFONT and its ISO variants), and the anti-aliased fonts that are loaded from disc. The normal font is quick and simple, but there is only one style. The anti-aliased fonts take time to load, and limit the palette available for other purposes in all but the 256 colour modes. Font Masher provides seventy-two varied fonts that are used in exactly the same way as BFONT, and which retain BFONT's great speed.

To make use of the module, type in the program in listing 1, and save it to disc. When run, it will assemble a module, and automatically save it to disc under the name RMmash. To use the module, just type:

```
*RMmash
```

If there is insufficient room in the RMA, type:

```
*Quit
```

```
*RMmash
```

```
*Basic
```

Now you can change fonts with the new star command *Mash. It takes a numeric parameter from 0 to 255, which defines the new font. *Mash 0 sets the font to normal. The table shows how the parameter is determined. If you want bold text, use:

```
*Mash 128
```

If you want bold and underlined, use:

```
*Mash 136
```

i.e. just add the codes for the features that you require.

Bit Number	Font Style	Decimal
0	Italics	1
1	Backward Italics	2
2	Feint	4
3	Underlined	8
4	Inverse	16
5	Superscript	32
6	Subscript	64
7	Bold	128

Table 1. Typestyles and corresponding parameters

There are three pairs of font styles that cannot be used together for obvious reasons. They are:

Italics and Backward Italics

Feint and Bold

Superscript and Subscript.

Having tried some of the new font styles, you will probably find a number which you want to use. To make reference to them easier, you can use the Arc's *SET Alias\$ command to give them appropriate names. For example, if you issue the following strings:

```
*SET Alias$Bold Mash 128
```

and

```
*SET Alias$Normal Mash 0
```

then every time that you type:

```
*Bold
```

the bold font will be engaged. The command:

```
*Normal
```

will reset the style to normal.

The program in listing 2 gives a short demonstration of some of the styles available.

NORMAL	ABC xyz 123
BOLD	ABC xyz 123
FEINT	ABC xyz 123
ITALIC	ABC xyz 123
UNDERLINED	ABC xyz 123
INVERSE ITALIC	ABC xyz 123
SUPERScript	ABC xyz 123

A display of the output from Listing 2

Listing 1

```
10 REM >Masher
20 REM Program New Font Styles
30 REM Version A 0.02
```

NEW FONT STYLES

```

40 REM Author      Peter Arnold
50 REM RISC User   March 1989
60 REM Program     Subject to copyright
70 :
80 DIM R% &1000
90 FOR I=4 TO 7 STEP 3
100 P%=0:O%=R%
110 [OPT I
120 EQU0:EQUD0:EQUB0:EQUD0
130 EQU0 title
140 EQU0 help
150 EQU0 command
160 .title
170 EQU0 "FontMasher"
180 EQU0 0
190 ALIGN
200 .help
210 EQU0 "Font Masher"+CHR$9+"1.00 (" +
MID$(TIME$,5,11)+")"
220 EQU0 0
230 ALIGN
240 .command
250 EQU0 "Mash"
260 EQU0 0
270 ALIGN
280 EQU0 start
290 EQU0 &00010001
300 EQU0 mashsyntax
310 EQU0 mashhelp
320 EQU0 0
330 .mashsyntax
340 EQU0 "Syntax: *Mash <0-255>"
350 EQU0 0
360 .mashhelp
370 EQU0 "*MASH with a number affects t
he characters shown on the screen. The e
ffects depend on the setting of the lowe
r eight bits of the parameter. Use *MASH
0 to reset the font."
380 EQU0 0
390 ALIGN
400 :
410 .start
420 STMFDR13!,{R0-R12,R14}
430 MOV R1,R0:MOV R0,#&30000000
440 SWI 33:BVS end
450 MOV R6,R2:MOV R0,#20
460 SWI 6:CMPI R6,#0
470 BNE carryon
480 B end
490 :
500 .carryon
510 MOV R3,#32
520 .getshape
530 ADR R1,block:STRB R3,[R1,#0]
540 MOV R0,#10:SWI 7
550 .go
560 AND R7,R6,#1:TEQ R7,#1
570 BNE go2:MOV R8,#1
580 MOV R9,#1:MOV R10,#31
590 .italics1
600 BL rotate:ADD R8,R8,#1
610 CMP R8,#4:BLT italics1
620 MOV R9,R10:MOV R8,#6
630 .italics2
640 BL rotate:ADD R8,R8,#1
650 CMP R8,#9
660 BLT italics2
670 .go2
680 AND R7,R6,#2:TEQ R7,#2
690 BNE go3
700 AND R12,R6,#1:CMPI R12,#1
710 BEQ go3:MOV R8,#1
720 MOV R9,#31:MOV R10,#1
730 .Bitalics1
740 BL rotate:ADD R8,R8,#1
750 CMP R8,#4:BLT Bitalics1
760 MOV R9,R10:MOV R8,#6
770 .Bitalics2
780 BL rotate:ADD R8,R8,#1
790 CMP R8,#9:BLT Bitalics2
800 .go3
810 AND R7,R6,#4:TEQ R7,#4
820 BNE go4
830 MOV R8,#1
840 .feint
850 LDRB R7,[R1,R8]
860 MOV R9,R7,ROR #1
870 AND R7,R7,R9
880 STRB R7,[R1,R8]
890 ADD R8,R8,#1:CMPI R8,#9
900 BLT feint
910 .go4
920 AND R7,R6,#8:TEQ R7,#8
930 BNE go5
940 .underline
950 LDRB R7,[R1,R8]
960 ORR R7,R7,#170
970 STRB R7,[R1,R8]
980 .go5

```

```

990 AND R7,R6,#16:TEQ R7,#16
1000 BNE go6:MOV R8,#1
1010 .inverse
1020 LDRB R7,[R1,R8]
1030 MVN R7,R7
1040 STRB R7,[R1,R8]
1050 ADD R8,R8,#1:CMP R8,#9
1060 BLT inverse
1070 .go6
1080 AND R7,R6,#32:TEQ R7,#32
1090 BNE go7
1100 MOV R8,#1:MOV R9,#1
1110 .superscript
1120 LDRB R10,[R1,R8]
1130 STRB R10,[R1,R9]
1140 ADD R8,R8,#2:ADD R9,R9,#1
1150 CMP R8,#9
1160 BLT superscript
1170 MOV R8,#5:MOV R9,#0
1180 .blankbottom
1190 STRB R9,[R1,R8]
1200 ADD R8,R8,#1:CMP R8,#9
1210 BLT blankbottom
1220 .go7
1230 AND R7,R6,#64:TEQ R7,#64
1240 BNE go8
1250 AND R12,R6,#32:CMP R12,#32
1260 BEQ go8:MOV R8,#7
1270 MOV R9,#8
1280 .subscript
1290 LDRB R10,[R1,R8]
1300 STRB R10,[R1,R9]
1310 SUB R8,R8,#2:SUB R9,R9,#1
1320 CMP R8,#1:BGE subscript
1330 MOV R8,#1:MOV R9,#0
1340 .blanktop
1350 STRB R9,[R1,R8]
1360 ADD R8,R8,#1:CMP R8,#5
1370 BLT blanktop
1380 .go8
1390 AND R7,R6,#128:TEQ R7,#128
1400 BNE define
1410 AND R12,R6,#4:CMP R12,#4
1420 BEQ define:MOV R8,#1
1430 .bold
1440 LDRB R7,[R1,R8]
1450 MOV R9,R7,ROR #1:ORR R7,R7,R9
1460 STRB R7,[R1,R8]:ADD R8,R8,#1
1470 CMP R8,#9:BLT bold
1480 .define

```

```

1490 MOV R8,#0:MOV R0,#23
1500 ADR R1,block:SWI 0
1510 .vdu23
1520 LDRB R0,[R1,R8]
1530 SWI 0
1540 ADD R8,R8,#1:CMP R8,#9
1550 BNE vdu23
1560 ADD R3,R3,#1:CMP R3,#126
1570 BLS getshape
1580 .end
1590 LDMFD R13!,{R0-R12,R14}
1600 MOV PC,R14
1610 .rotate
1620 LDRB R7,[R1,R8]
1630 MOV R7,R7,ROR R9
1640 STRB R7,[R1,R8]
1650 MOV PC,R14
1660 .block
1670 EQU 0:EQU 0:EQU 0
1680 ]:NEXT
1690 OSCLI"SAVE Rmmash "+STR$-R%+"+"+ST
R$~P%
1700 OSCLI"SETTYPE Rmmash FFA"

```

Listing 2

```

10 REM >Mash2Demo
20 REM Font Style Demonstration
30 :
40 *RMMASH
50 ON ERROR REPORT:IF ERR=17 THEN OSC
LI("MASH 0"):END ELSE PRINT" at line ";E
RL:END
60 MODE 0
70 *MASH 0
80 PRINT"NORMAL ABC xyz 123"
90 *MASH 128
100 PRINT"BOLD ABC xyz 123"
110 *MASH 4
120 PRINT"FEINT ABC xyz 123"
130 *MASH 1
140 PRINT"ITALIC ABC xyz 123"
150 *MASH 8
160 PRINT"UNDERLINED ABC xyz 123"
170 *MASH 19
180 PRINT"INVERSE ITALIC ABC xyz 123"
190 *MASH 32
200 PRINT"SUPERSCRIPT ABC xyz 123"
210 *MASH 0

```