

# *S*PRITES

PREMIER MICROSYSTEMS  
LONDON ENGLAND

## COPYRIGHT NOTICE

The Sprite Board was designed by Justin Johnson. The manual was written by Justin Johnson with formatting and editing by John King. No part of the operating software may be stored or copied or reproduced by any means whatsoever other than for the personal use of the original retail purchaser without the written consent of PREMIER MICROSYSTEMS LTD.

No part of this manual may be reproduced for any reason whatsoever (except very brief excerpts for review purposes) without the written permission of PREMIER MICROSYSTEMS LTD.

Every copy of the Sprite operating software is uniquely coded and any backup will contain this coding. PREMIER pursues an active policy against abusers of our copyrights.

First Published January 1984 1983

Published for Premier Microsystems LTD by Premier Publications.

ISBN 1 85025 003 0

COPYRIGHT 1984 PREMIER MICROSYSTEMS LTD

## SPRITE KIT

Please read this page BEFORE you start building the kit.

This kit is complex in nature and the board is tightly packed. If on receipt of the kit you feel that you are not competent enough to undertake construction AND possible fault finding, please return the kit to us (enclosing the difference in price between built and kit form and return postage) and we will return it to you as a built and fully tested unit. This offer is open for three calendar months from the date of purchase.

PLEASE NOTE THAT IF CONSTRUCTION IS UNDERTAKEN AND THE UNIT FAILS TO FUNCTION, A REPAIR CHARGE OF £25.00 PER HOUR WILL BE MADE WITH A MINIMUM OF ONE HOUR CHARGED. We regret making this condition, but long experience has shown that badly built boards are notoriously difficult to fault-find and are often subsequently unreliable - it's often quicker for us (and cheaper for the customer) to scrap the board and start again!

\*\*\*\*\*

Main Computer Board..... Clock Speed.

The SPRITE GRAPHICS BOARD will NOT, repeat NOT run at any speed other than 1mhz as the TEXAS TMS 9929A is specifically designed to run at that speed. CIE customers, and UK101/CIU customers who have modified their boards to run at 2mhz (or any speed  $< >$  1mhz), are advised to either make the facility switchable or hardwire back to 1mhz. We cannot give information regarding conversion back to 1mhz as there are many ways of achieving it in the first place!

If you are unsure whether your computer runs at 1 or 2mhz - try timing the following line of BASIC

```
10 FOR N = 1 TO 10000 : NEXT
```

If the loop takes around 11 seconds to run, you have a 1mhz machine. A 2mhz machine will take only half the above time.

*The Dragon sprite code automatically turns off the speed up (poke 1HFFD7) if used in a program.*

### ASSEMBLY & ADJUSTMENT

#### A/ ASSEMBLY

The following stages of construction are recommended.

- 1/ Fit all resistors
- 2/ Fit IC sockets
- 3/ Fit capacitors & diodes (Note C31 is next to TR4 and is unmarked)
- 4/ Fit IC1 & IC2

At this point the PCB should be checked for solder bridges and bad joints.

Provision has been made for both + 5v and + 12v to be supplied via the J1 connector (the other voltages required are generated on the Sprite board). + 5v can be supplied via pin 11 of J1 and the + 5v link (inboard from IC7) should be made if this is required.

+ 12v can be supplied via pin 10 of J1, and the + 12v link (between IC3 & J1) should be made if this is required (pin 10 on UK101/SB J1 is normally ground and should not be used unless special provision has been made on your UK101/SB)

The PCB can now be powered up and the correct voltages checked for.

+12v	ON PINS 8	OF IC11 - IC18
-5v	ON PINS 1	OF IC11 - IC18
+5v	ON PINS 9	OF IC11 - IC18
0v	ON PINS 16	OF IC11 - IC18
+11.5v	ON PIN 16	OF IC22
+5v	ON PIN 1	OF IC21

If the above voltages are correct the power should be removed and the remaining components fitted (Transistors, Crystals, Modulator and ICs). Link K - R should be cut and link K - U made (this moves the base address from \$8000 to \$E000).

The Sprite board is now ready for adjustment.

## B/ ADJUSTMENT

Connect the Sprite board to the UK101/SB via the J1 cable and power up.

RESET C (Normal cold start)  
RESET M  
.E002G

Enter the following program

```
5 CLS15 : REM CLEAR BORDER TO WHITE
10 MODE4 : REM GOTO MULTICOLOUR
20 FOR X = 0 TO 63 : C = INT(X/4)
30 LINE X,0,X,47,C : NEXT X
40 GOTO 40
RUN
```

10 PMODE4 : REM DRAGON

This program will set up a colour bar pattern on the processor, the following adjustments are made to bring up the full colour picture. If no picture is present after initialisation turn off and check your construction.

1/ set CV1 & CV2 so that the moving vanes are two thirds meshed with the fixed vanes.

2/Wind RV1 fully clockwise, a faint clicking will be heard at the end of the travel.

Make 12 full turns anticlockwise. Gently adjust CV1 and RV1 for a stable colour picture. If you have difficulty, check the tuning of your television set (if possible turn off the AFT/AFC).

CV2 can be gently adjusted to give clear vertical separation between the colour bars.

It is recommended that a non conductive trimming tool is used when adjusting CV1 and CV2.

When the above program is run prior to adjustment some random patterning may be seen ! After the above adjustments have been made use CONTROL C and re-run the program to confirm the adjustment.

## NOTES FOR CEGMON S *(UK101 only)*

The scroll width is set at location \$DF (223 decimal). For a normal UK101 this is set to \$40 (64 decimal). This will allow your machine to run your old software on the original screen. When the Sprite board is initialised the Cegmon S locations in page 2 and \$E0 are set up for the Sprite screens. The settings have to be chosen to give a small border both sides of the text. This is necessary as domestic television sets are always adjusted to overscan.

	MODE 1	MODE 2	
\$DF (223)	\$28 (40)	\$20 (32)	SCROLL WIDTH
\$0222 (546)	\$25 (37)	\$1B (27)	COLUMN WIDTH -1
\$0223 (547)	\$01 (01)	\$02 (02)	LOW BYTE TOP
\$0224 (548)	\$D0 (208)	\$D0 (208)	HIGH BYTE TOP
\$0225 (549)	\$99 (153)	\$E2 (226)	LOW BYTE BASE
\$0226 (550)	\$D3 (211)	\$D2 (210)	HIGH BYTE BASE

## MODIFICATIONS TO VORTEX AND ROMDOS

To enable Vortex or Romdos to operate correctly with the Sprite board and Cegmon S, carry out the modifications below. These must be carried out before fitting Cegmon S.

```
1/ !"CALL 3000=02,1
2/ POKE12511,64 (screen width)
3/ !"SAVE 02,1=3000/1
```

### VORTEX

```
4/ !"CALL 3A00=05,1"
5/ POKE16472,64
6/ !"SAVE 05,1=3A00/8"
```

### ROMDOS

```
4/ !"CALL 2A00=01,1"
5/ POKE12376,64
6/ !"SAVE 01,1=2A00/8"
```

To enable the Cegmon linker to work with OS65D, add the following.

POKE223,64 : POKE12376,64

Please note that the Sprite Board will NOT work with OS65D, as the Sprite Board uses BASIC in ROM, and under OS65D the BASIC is located in RAM at a totally different address.

## COMPONENT LIST

### RESISTORS $\frac{1}{4}w$ 5%

4k7	x 7	R1,R2,R3,R10,R13,R21,R27	YELLOW VIOLET RED
10k0	x 4	R4,R18,R19,R29	BROWN BLACK ORANGE
330R	x 2	R7,R16	ORANGE ORANGE BROWN
560R	x 4	R6,R8,R11,R12	GREEN BLUE BROWN
1k0	x 6	R9,R20,R25,R26,R34	BROWN BLACK RED
2k2	x 6	R14,R15,R17,R22,R31,R32	RED RED RED
4M7 ( $\frac{1}{2}w$ )	x 1	R23	YELLOW VIOLET GREEN
3k3	x 1	R24	ORANGE ORANGE RED
75R	x 2	R28,R30	VIOLET GREEN BLACK
270R	x 1	R33	RED VIOLET BROWN
390R	x 1	R5	ORANGE WHITE BROWN

### RESISTORS VARIABLE

2k0 RV1

### CAPACITORS VARIABLE

5p5 - 65pf CV1, CV2

### CAPACITORS

33pf	x 4	CERAMIC	C1,C2,C29,C30
10 $\mu$ f 16v	x 6	ELECTROLYTIC	C3,C6,C7,C22,C25,C32
47 $\mu$ f 16v	x 3	ELECTROLYTIC	C4,C5,C10
220nf	x 1	POLYESTER	C8
470nf	x 1	POLYESTER	C9
100nf	x 26	CERAMIC	C11 - C14,C16 - C18,C20,C24,C26,C27,C31,C33 - C46
150pf	x 2	POLYSTYRENE	C15,C19
22 $\mu$ f 10v	x 1	ELECTROLYTIC	C23
10nf	x 1	CERAMIC	C28
220pf	x 1	POLYSTYRENE	C21

NOTE 100nf may be marked 0.1 $\mu$ f  
10nf may be marked 0.01 $\mu$ f

### DIODES

BAW 62 D1  
BZY 88 7v5 (ZENER) D2  
BZY 88 5v1 (ZENER) D3,D4

NOTE Band around diode indicates positive or + end

### CRYSTALS

4.433Mhz X1  
10.7Mhz X2

## MODULATOR

UM 1233

M1

## TRANSISTORS

BC 477

TR1

BC107

TR2, TR4, TR5

BC109

TR3

NOTE The case of TR5 must NOT touch the modulator case.

## INTEGRATED CIRCUITS

78L05

IC1

7660

IC2

74LS00

IC3

74LS20

IC4

74LS08

IC5

74LS138

IC6

74LS25

IC7

74LS27

IC8

TMS 9929A

IC9

2764 EPROM

IC10

4116

IC11 - IC18

7416

IC19

4066B

IC20

74LS74

IC21

LM 1889

IC22

## HARDWARE

40 PIN DIL SOCKET

2

28 PIN DIL SOCKET

1

18 PIN DIL SOCKET

1

16 PIN DIL SOCKET

9

14 PIN DIL SOCKET

8

8 PIN DIL SOCKET

1

40 WAY HEADER CABLE ASSY

1

TERMINAL PINS

7

SPRITE PCB

1

## OPTIONAL HARDWARE

P.S.U.

TO PROVIDE +5v AND +12v

CASE

FOR SPRITE BOARD AND P.S.U

## TOOLS REQUIRED FOR CONSTRUCTION

A fine tipped soldering iron, 15 watts maximum (IMPORTANT!!)

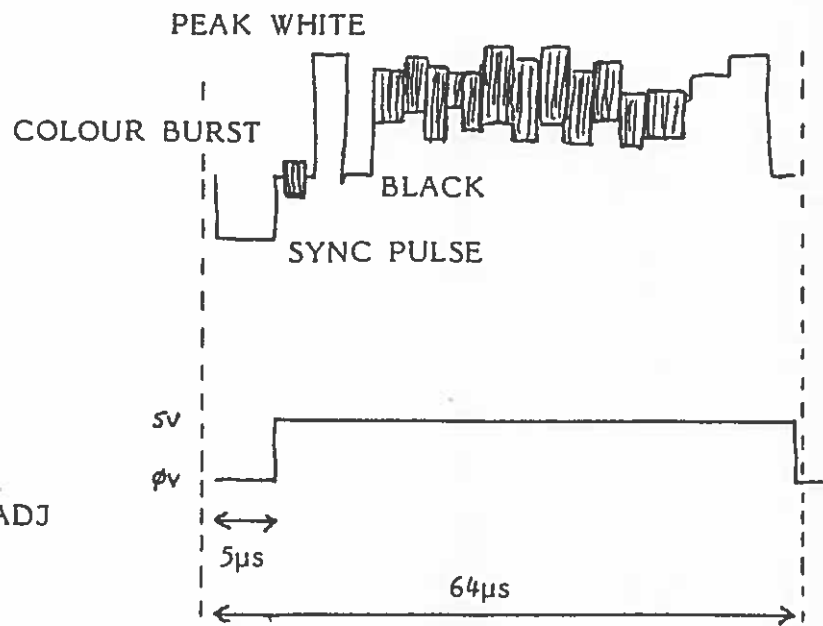
A small pair of side cutters

A small pair of fine snipe nose pliers

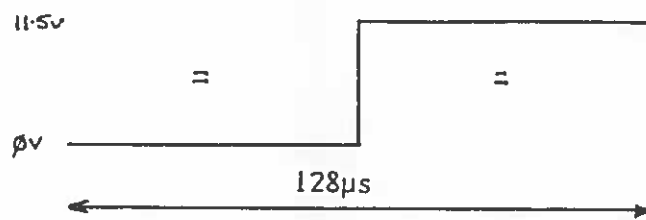
A reel of fine solder

A multi-meter for checking voltages and solder bridges

COMPOSITE VIDEO O/P

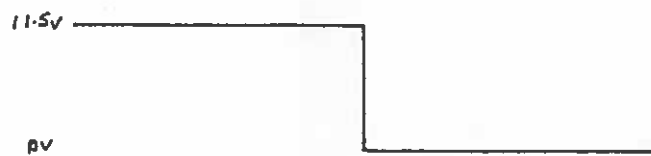


IC 19 PIN 10  
SYNC STRIPPER RV1 ADJ

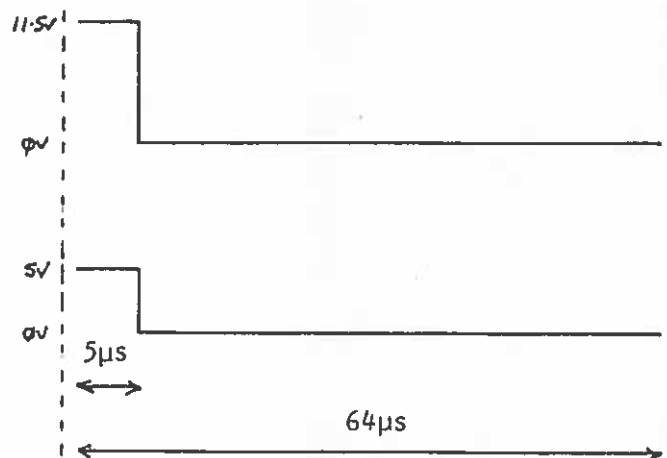


IC 20 PIN 5

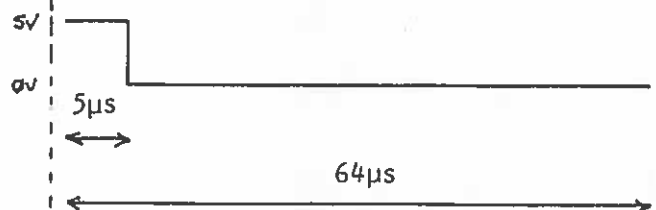
IC 20 PIN 13



IC 20 PIN 6



IC 21 PIN 3

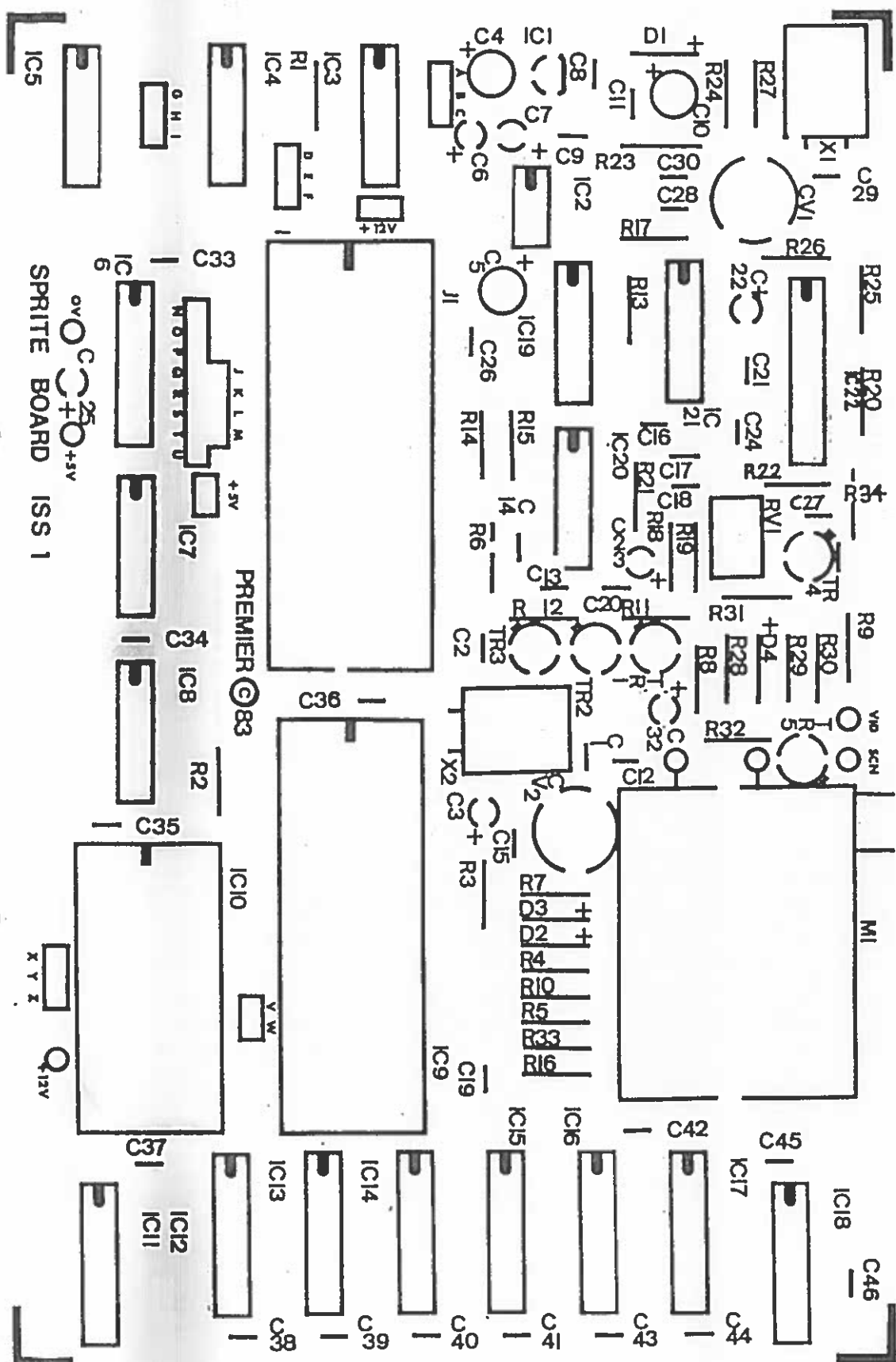


IC 22 PIN 17



ADJ CV1 TO GIVE PEAK AMPITUDE





# *Sprites*

PREMIER MICROSYSTEMS  
LONDON ENGLAND

# CONTENTS

## Page No.

INTRODUCTION	5 ✓
INITIALISING THE SYSTEM	6 ✓
MODE SELECTION	8 ✓
COLOURS	9 ✓
CLEAR SCREEN	9 ✓
CHARACTER COLOUR & BACKGROUND	10 ✓
PATTERN PLANE COMMANDS	11 ✓
PRINT @	11 ✓
PLOTTING IN COLOUR	12 ✓
PLOT/SET (MODES 3 & 4 ONLY)	12 ✓
UNPLOT/RESET (MODES 3 & 4 ONLY)	12 ✓
LINE	13 ✓
DRAGON HIRES	13 ✓
CIRCLE	14 ✓
UK101/OHIO SPECIAL FACILITIES	15 ✓
SPRITE CHARACTER SETS	16 ✓
USING SPRITES	16 ✓
CHANGING SHAPE	18 ✓
8 x 8 PIXEL CHARACTER SET	18
16 x 16 PIXEL SHAPES	20
ENTRIES INTO SPRITE ROM	21
SYNTAX SUMMARY	23

PIERS LETCHER

PEL Small Computer NEWS

VNU

62 OXFORD ST

636-6890 FX264

Press release  
info

Dragon sprites For review

## INTRODUCTION

The Premier Microsystems Sprite graphics board is an implementation of the Texas Instruments TMS 9929A Video Display Processor (V.D.P.). In addition to the processor, the board has 16K of dynamic RAM, address decoding, a full PAL encoder and a ROM containing the system software. (A ROM is not used on the VG/TRS80 systems). The output from the board is suitable for a normal U.H.F colour television or a PAL monitor.

The board has four distinct modes of operation.

1	Text	40 x 24	Text only
2	Graphics	32 x 24	Text & Sprites
3	Hi Res	256 x 192	High resolution screen & Sprites
4	Multicolour	64 x 48	Chunky full colour graphics & Sprites

A total of 15 colours are available for use in all these modes. The display consists of 34 planes, one above the other. The lowest plane is the background colour, the next plane is the pattern or text plane. All text, Hires and chunky patterns appear on this level. The pattern plane is identical in principle to the normal display on a non Sprite computer. The 32 Sprite levels are located above the pattern plane. When the Sprites are not in operation these planes are transparent (COLOUR 0), providing an uninterrupted view of the pattern plane.

Each of the Sprite planes can contain 1 Sprite, which can be positioned to the nearest pixel on the screen. These Sprites may be in any of the 15 available colours. Three sizes of Sprite are available:- 8 x 8, 16 x 16, 32 x 32. However all Sprites displayed have to be within the same size range.

All text and Sprites are stored in the systems memory. This allows all the shapes to be modified at will. The system software loads a full set of text, graphics, and Sprite characters whenever a mode change takes place. Command words have been provided to alter these shapes if so desired.

The system software has been developed to be totally integrated with the host computers BASIC in ROM. This has meant that command words and facilities will vary from implementation to implementation. DRAGON users will find that the new commands are extensions of existing graphics commands, while UK101, and VG/TRS80 users will find a complete new set of command words.

Although the Sprite Graphics board has 16K of memory, this is external to the computer's main memory map, so users do not need to worry about contention of memory space. In the 6502 and 6809 systems the video processor occupies only two memory locations, usually \$E000 and \$E001. On Z80 systems the processor uses two I/O ports.

A full technical specification of the TMS 9929/A chip is available from

TEXAS INSTRUMENTS  
MARTIN LANE  
BEDFORD

## INITIALISING THE SYSTEM

### DRAGON

- 1/ Turn off the computer and insert the connector into the cartridge slot
- 2/ Power up the system and wait for the normal start up message
- 3/ Type EXEC&HE000 then hit ENTER
- 4/ Connect colour TV/Monitor to the Sprite Board colour output

The screen should display the startup message in white on a dark blue background and will work in MODE 1 (40 x 24) until instructed otherwise.

### UK101/SUPERBOARD

(Ensure that the board is properly connected via the J1 socket and CEGMON 'S' has been installed. The board will NOT operate without CEGMON 'S'.)

- 1/ Turn on the computer, then RESET C to fire up normal BASIC. At this point the machine may be used as before to run existing software.

#### ROM Sprite Software

- 2/ RESET 'M'
- 3/ E002 G

#### DISK Sprite Software \*

- 2/ RESET 'D'
- 3/ POKE11,2:POKE12,224:X=USR(X)
- 4/ Connect colour TV/Monitor to colour output

The screen should display the startup message in white on a dark blue background. The board will now work in MODE 1 (40 x 24) until instructed otherwise.

(\* Disk-based UK101/OHIO systems MUST use ROMDOS or VORTEX as the system requires BASIC in ROM)

### VIDEO GENIE/TRS80

The design of these machines prevents the use of a ROM based controller. The software has been provided on a tape or disk. Assuming the board has been connected to the system

- 1/ Turn on the computer and initialise BASIC (ROM or DISK)
- 2/ Load system software from tape or disk

3/ Execute software as instructed on tape or disk

4/ Connect colour TV/Monitor to colour output

The screen should display the startup message in white on a dark blue background. The board will now work in MODE 1 (40 x 24) until instructed otherwise.

## MODE SELECTION

### SYNTAX

MODE N                    (UK101 & VG/TRS80)

Pmode N                  (DRAGON)

MODE 1                  Is the 40 x 24 text mode. This is the mode to use when programming, as it gives the largest number of characters on the screen. Sprites are not available in this mode. Only two colours are available at any one time in this mode.

MODE 2                  Is the 32 x 24 text/graphics mode. This mode allows text to be mixed with graphics and Sprites. A full range of colours and facilities are available.

MODE 3                  Is the 256 x 191 Hires mode. This gives high resolution colour graphics and Sprites ~~(All existing hires graphics commands are available for DRAGON users).~~

MODE 4                  Is the 64 x 48 chunky colour graphics mode. This mode gives vivid blocks of colour with Sprites.

Users should note that whenever a mode change takes place the V.D.P and video RAM are re-initialised. This is necessary due to different memory allocations in different modes. As a result all text and graphics characters will revert to their original shapes, and colours.

Not all new command words are available in all modes, so if in doubt check before use. Whenever a PRINT, PRINT@ or PUTAT command is encountered the system will automatically revert to the last text mode used.

## COLOURS

A number of commands have been included to control the colours displayed on the screen.

### COLOUR

Changes the background colour B and character colour F.

#### SYNTAX

COLOUR B,F	(UK101, VG & TRS80)	B=Background
COLOR <del>B,F</del> F,B	(DRAGON)	F=Foreground

The colours available are:-

0	Transparent (Not normally used)
1	Black
2	Medium Green
3	Light Green
4	Dark Blue
5	Light Blue
6	Dark Red
7	Cyan
8	Medium Red
9	Light Red
10	Dark Yellow
11	Light Yellow
12	Dark Green
13	Magenta
14	Grey
15	White

NB. These are the colours displayed on a studio quality PAL monitor. Some variations may be seen on a domestic colour TV. Under no circumstances attempt to alter your TV's internal controls.

### CLEAR SCREEN

#### SYNTAX

CLS N	(UK101, DRAGON)
CLV N	(VG & TRS80)

Clears the screen and changes the background to colour N. The choice of colours is as in the COLOUR command. This also resets all Sprites currently displayed.

Example:- CLS12 changes background to Dark Green.  
CLV12



## CHARACTER COLOUR & BACKGROUND

The V.D.P allows the user to change the individual colour background and character colours. This facility is only available in MODE 2. Any combination of colours is permitted, (although some combinations look 'Yeuch'). The only restriction is that character colours can only be altered in groups of 8.

ie If you change character 64, then all characters from 64 to 71 will change, or change character 3 and characters 0 to 7 will change.

### SYNTAX

INK C,B,F (UK101, VG & TRS80)  
COLOR! C,B,F (DRAGON)

C = Character number    0 <= C <= 255  
B = Background        0 <= B <= 15  
F = Foreground        0 <= F <= 15

Colour set as in the COLOUR command.

Try the following.

UK101 & VG

MODE2  
INK72,7,1  
LIST

DRAGON

PMODE2  
COLOR!72,7,1  
LIST

## PATTERN PLANE COMMANDS

This section deals with those commands that affect the pattern/text plane of the display. The facilities available depend on the host computers existing arrangements. The operating system has been written so that none of the computers original facilities have been lost.

### PRINT @ (MODES 1 AND 2 ONLY)

#### SYNTAX

PRINT@ N,A\$  
PRINT@ N,A

DRAGON & VG These commands operate as normal but the range of the PRINT@ command has been modified to suit the mode in use.

PUTAT X,Y,A\$  
PUTAT X,Y,A

UK101,OHIO X and Y define the position of the printed string. X is the horizontal location, Y is the vertical.

$0 \leq X \leq 39$  in mode 1 and  $0 \leq X \leq 31$  in mode 2.

$0 \leq Y \leq 23$  in both modes.

X & Y are measured from the top left of the screen.

## PLOTTING IN COLOUR

Two commands have been incorporated to allow coloured pixels to be plotted into the pattern plane. These commands are available in the multicolour and Hires modes. In multicolour mode (4) there is no restriction on the colours available for each pixel. In Hires mode (3) much smaller pixels are used. The TEXAS V.D.P restricts the horizontal definition of coloured pixels. The processor changes pixels in horizontal groups of eight. If any of the pixels is changed every pixel in the group changes. Therefore in any group of eight horizontal pixels, all pixels will be the same colour, but may be on or off in any combination. A little experience will show that this is not normally a major restriction.

### PLOT/SET (MODES 3 & 4 ONLY)

Used to put a pixel of colour C at horizontal position X and vertical position Y.

#### SYNTAX

PLOT X,Y,C (UK101 & VG)

SET X,Y,C (DRAGON) *PMODE 4*  
~~PSET X,Y,C~~ " *PMODE 3*

In MODE 3

$0 \leq X \leq 255, 0 \leq Y \leq 191$  C as in COLOUR command.

In MODE 4

$0 \leq X \leq 63, 0 \leq Y \leq 47$  C as in COLOUR command.

### <sup>PRESET</sup> UNPLOT/~~RESET~~ (MODES 3 & 4 ONLY)

Removes a pixel from the display. X & Y are as in PLOT/SET command.

#### SYNTAX

UNPLOT X,Y (UK101 & VG)

~~RESET X,Y~~ (DRAGON)

SET X,Y,B (DRAGON, B = current background in *PMODE 4*)  
PRESET X,Y (DRAGON *PMODE 3*)

## LINE

Draws a line from position (X,Y) to position (X1,Y1) in colour C. The parameters allowed for X,X1,Y,Y1 are as in the PLOT command.

MODES 3 & 4 ~~DK101 & VG ONLY~~

~~DRAGON USERS SHOULD CONSULT THE SPECIAL DRAGON HIRES SECTION BELOW~~

## SYNTAX

LINE X,Y,X1,Y1,C

*all machines*

where X,X1,Y,Y1 are as defined above.

To 'undraw' a line:-

LINE\* X,Y,X1,Y1 (*DK101 / VG*)  
LINE# X,Y,X1,Y1 (*DRAGON*)

~~LINE X,X1,Y,Y1~~

This is used to remove a line from (X,Y) to (X1,Y1).

## DRAGON HIRES

The DRAGON computer has a MICROSOFT extended colour BASIC. This has been integrated with the Hires mode on the Sprite board (PMODE3). All DRAGON Hires commands now work in the same way as on the normal highest resolution screen (the original PMODE4,1). These commands are only available in Hires. The only command which cannot be used is PAINT. In Hires the DRAGON uses the original Hires memory. This memory cannot be used for program space if it is intended to employ PMODE3.

## UK101/OHIO SPECIAL FACILITIES

### CEGMON and COPY

The UK101 operating system is closely linked into the special CEGMON S monitor. All the screen editing facilities of CEGMON are available on the system. Many users will have BASIC 5 or other graphics aids. A special command has been incorporated to allow BASIC 5 to work on the Sprite board. When using a BASIC 5 command follow with the special word COPY. COPY mirrors the contents of the original screen memory into the Sprite memory.

eg

&SCR33:COPY

Graphics can be POKEd to the original screen and then COPYed across to the Sprite pattern plane. When operating, the Sprite graphics board uses the original screen memory as a special backing store. Do not attempt to use this area for other storage.

When the UK101 is RESET the screen reverts to the old screen display. RESET W will return to the new screen. The monitor is now accessed by the PANEL command NOT RESET M. All the CEGMON monitor features are still available. Return to BASIC with RESET W or .0000G.

## USING SPRITES

Sprites are special animation shapes which provide very smooth motion and multi-level pattern overlay. The location of each Sprite is defined by the position of its top left hand corner. The Sprite can be moved on a pixel by pixel basis. Each of the 32 Sprite planes can hold 1 Sprite of any colour. The TEXAS V.D.P only permits 4 Sprites on any horizontal line. The highest planes have the highest priority. Unused Sprites are located at the position 0,0 and are transparent in colour. When using Sprites always start at level 0 and work down. Do not use lower priority planes if any higher plane is unused!

Two sets of Sprite shapes are available, all of which may be redefined. The first (small) set is the entire character set, (only available in MODE 2). The second (large) set is a special set of 32 gaming characters (available in MODEs 2,3,4). Each set has two sizes. The small set can be 8 x 8 (normal character size) or 16 x 16. The large set can be 16 x 16 or 32 x 32.

The type of Sprite displayed is controlled by the SIZE command.

### SYNTAX

SIZE A,B — UK101 VG  
SCREEN A,B — DRAGON

A gives the character set, B gives the magnification.

SIZE 0,0	Small set	8 x 8 pixels	MODE 2 only
SIZE 0,1	Small set	16 x 16 pixels	MODE 2 only
SIZE 1,0	Large set	16 x 16 pixels	MODEs 2,3 & 4
SIZE 1,1	Large set	32 x 32 pixels	MODEs 2,3 & 4

Note that the V.D.P can only display one size of Sprite at any one time.

The Sprites can be removed by use of the CLS/CLV command.

Sprites are moved using a special command. This command positions a particular Sprite on its chosen level and automatically erases its previous position. It is NOT necessary to clear the Sprite from its original position when moving it - this allows fast and smooth graphics to be achieved from BASIC.

### SYNTAX

SPRITE X,Y,C,S,L (UK101 & VG)  
PUT@ X,Y,C,S,L (DRAGON)

0 <= X <= 255	Horizontal position of Sprite.
0 <= Y <= 191	Vertical position of Sprite.
0 <= C <= 15	Colour of Sprite (as COLOUR command)
0 <= S <= 255	SIZE 0 Shape of Sprite
0 <= S <= 31	SIZE 1 Shape of Sprite
0 <= L <= 31	Level or priority of Sprite, 0 is the highest priority

Remember that each level can only hold one Sprite. A Sprite can be removed from the display by setting its colour to transparent.

ie SPRITE0,0,0,0,0,3 turns off the Sprite on level 3.

## CHANGING SHAPE

All character shapes are held within the V.D.P Ram area. Therefore all characters may be re-defined to any shape required for a particular program. The new word SHAPE/PRINT! is used to re-define a character. It should be noted that all shapes revert to original on mode change.

### 8 x 8 PIXEL CHARACTER SET (NORMAL SIZE ALPHANUMERIC ASCII)

#### MODES 1 & 2 ONLY

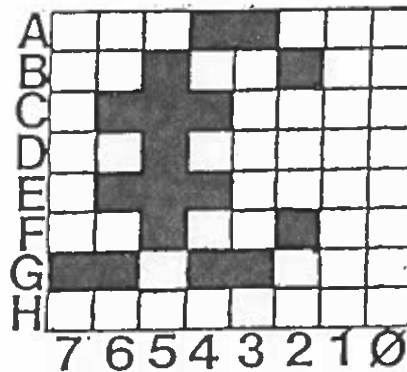
#### SYNTAX

SHAPE N,A,B,C,D,E,F,G,H (UK101 & VG)

PRINT! N,A,B,C,D,E,F,G,H (DRAGON)

N is the ASCII character code (0 - 255) and A - H define each line of the character.

Each character is made up from an 8 x 8 matrix, the following example shows the technique.



Columns 0 & 1 are not displayed in MODE 1

Column 7 and row H are not normally used for text characters, to allow spaces between letters. All rows and columns may be used for graphics if required. In MODE 1 the V.D.P DOES NOT display columns 0 & 1. These should not be used when changing characters for this mode.

Each row of the matrix can be represented by an 8 bit number. A '0' represents background and a '1' foreground. These numbers are used in the SHAPE/PRINT! command.

~~=====~~

The shape can be drawn out on a grid as shown in the diagram. The shape must now be converted into numbers

Row A	00011000	= 24
Row B	00100100	= 36
Row C	01110000	= 112
Row D	00100000	= 32
Row E	01110000	= 112
Row F	00100100	= 36
Row G	11011000	= 216
Row H	00000000	= 0

The hash # sign is ASCII character 35, we can now re-define the shape.

SHAPE 35,24,36,112,32,112,36,216,0	(UK101 & VG)
PRINT! 35,24,36,112,32,112,36,216,0	(DRAGON)
PRINT CHR\$(35)	(All)

The new character will also be generated when SHIFT 3, the # sign, is depressed.

This advanced feature can be used to design any type of gaming character or character set. Special national language characters like the pound sign are easy to introduce into the character set.



# 16 x 16 PIXEL SHAPES

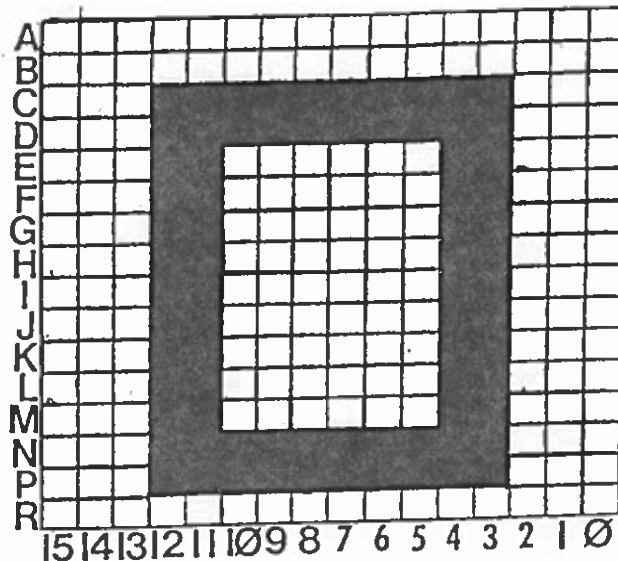
## MODES 2, 3 & 4

### SYNTAX

SHAPE\* N,A,B,C,D,E,F,G,H,I,J,K,L,M,N,P,R  
~~PRINT#~~ N,A,B,C,D,E,F,G,H,I,J,K,L,M,N,P,R  
 RESET

(UK101,TRS,VG)  
 (DRAGON)

N is the Sprite shape number (0 - 31)  
 A - R define the lines of the shape (0 - 65535)  
 Each big Sprite is defined on a 16 x 16 matrix



The principle is identical to that outlined in the small shape re-definition. Large Sprites are more complicated because 16 bit binary numbers are used (0 - 65535).

Example Produce an open box character for Sprite shape 4.

Row A	0000 0000 0000 0000	= 0
B	0000 0000 0000 0000	= 0
C	0001 1111 1111 1000	= 8184
D	0001 1111 1111 1000	= 8184
Rows E-M	0001 1000 0001 1000	= 6168
N & P	0001 1111 1111 1000	= 8184
Row R	0000 0000 0000 0000	= 0

SHAPE\* 4,0,0,8184,8184,6168,6168,6168,6168,6168,6168,6168,6168,8184,8184,0.

On the UK101, which only has a 71 character input buffer, line length can be a problem so use:-

A = 0 : B = 8184 : C = 6168

SHAPE\* 4,A,A,B,B,C,C,C,C,C,C,C,C,B,B,A (UK101)

RESET  
~~PRINT#~~ 4,0,0,8184,8184,6168,6168,6168,6168,6168,6168,6168,6168,8184,8184,0  
 (DRAGON)

# SYNTAX SUMMARY and examples.

Command	DRAGON	TRS/VG	UK101/OHIO
MODE	-	MODE3	MODE3
PMODE	PMODE3	-	-
COLOUR	-	COLOUR4,5	COLOUR4,5
COLOR	COLOR4,5	-	-
CLS	CLS4	-	CLS4
CLV	-	CLV4	-
INK	-	INK200,4,7	INK200,4,7
COLOR!	COLOR!200,4,7	-	-
PRINT@	PRINT@347,"TEXT";	PRINT@347,"TEXT"	-
PUTAT	-	-	PUTAT20,15,"TEXT"
PLOT	-	PLOT147,45,3 <del>X</del>	PLOT147,45,3 <del>X</del>
PSET	<del>PSET</del> 147,45,3 <del>X</del>	-	-
SET	<del>SET</del> 47,45,34 SET 30,24,3	-	-
UNPLOT	-	UNPLOT147,45,34 <del>X</del>	UNPLOT147,45,34 <del>X</del>
RESET	RESET147,45,34	-	-
LINE	LINE 2,6,28,34,12	LINE2,6,28,34,12	LINE2,6,28,34,12
LINE*	-	LINE*2,6,28,34,12 <del>X</del>	LINE*2,6,28,34,12 <del>X</del>
LINE#	LINE# 2,6,28,34	-	See text
COPY	-	-	See text
SIZE	<del>SIZE1,0</del>	SIZE1,0	SIZE1,0
SCREEN	SCREEN 1,0	-	-
SPRITE	-	SPRITE120,49,4,1,13	SPRITE120,49,4,1,13
PUT@	PUT@120,49,4,1,13	-	-
SHAPE	-	(See text)	(See text)
SHAPE*	-	(See text)	(See text)
PRINT!	(See text)	-	-
RESET	(See text)	-	-
PRINT!!	(See text)	-	-
CIRCLE	CIRCLE 128,96,6,3,2	-	-

## Sprite character sets.

Various Modes. The system software provides character sets for the character sets. In MODE1 the text mode, ~~the~~ character 0-127 are the standard ASCII character set and characters 128-255 are the same characters in inverse.

In MODE2 characters 0-127 are still the ASCII character set but characters 128-255 are now coloured graphics blocks. To view the character sets enter the following simple program:

```
10 CLS : REM CLV U.G.  
20 INPUT "MODE"; M : IF M < 0 OR > 2 THEN 20  
30 : MODE M : REM " PMODEM DRAGON"  
40 FOR J = 0 TO 255 : REM CHARS < 14 ARE CONTROL CHARS  
50 PRINT CHR$(J); " "  
60 NEXT  
70 END
```

All these shapes can be altered using the appropriate commands. These shapes are also used for the small sprites. Shapes 0 to 31 are supplied for large sprites. REMEMBER a Mode change restores all shapes to the standard set.

## Hi-res

The new high res facilities provided by the Sprite board replace those normally available on the Dragon. The original BASIC commands cannot be harnessed to allow the new system to use all of its colours in high resolution, we have provided new commands to enable the user to make the most of the new display. However because of the very different operation of the Sprite display processor it has not been possible to duplicate all of the Dragon commands.

The first major difference is that the new processor requires very little of the system memory. The new code frees the original graphics memory for program space. (Try PRINT MEM after start up!). You will also notice the changed ~~sprite~~ syntax on the Hi-res commands. Whilst not as comprehensive as the Dragon commands, the wider range of colours (15 instead of 2) and sprites give a much more striking and vivid display.

[It should be ~~noted~~ remembered that the redefinable and colourable graphics in MODE2 can also be used to construct Hi-res displays]

LINE		LINE#	see	main	Text
------	--	-------	-----	------	------

PSET		PRESET	"	"	"
------	--	--------	---	---	---

PCLS	now identical to	CLS	as in	main	Text
------	------------------	-----	-------	------	------

PCLR	as before,	but	boots up as	PCLR	0
------	------------	-----	-------------	------	---

(Memory location 110600 → 00FF reserved for system use)

SCREEN	see	sprites
--------	-----	---------

PUT	"	"
-----	---	---

DRAW, GET, PAINT, PCOPY	Not available
-------------------------	---------------

CIRCLE

PROCES 3 & 4 (PC error in after Maths!)

SYNTAX

CIRCLE X, Y, C, R[, S, F][; H, W] ([ optional ])

X =	X coord of Centre	range as for PSET & SET
Y =	Y " " "	" " " "
C =	Colour	range as COLOR
R =	Radius	$5 \leq R \leq 127$
S =	start	$0 \leq S \leq 119$
F =	Finish	$5 \leq F \leq 119$
H =	Height	$0 \leq H \leq 7$ (NOTE SEMI CIRCLE IN SYNTAX)
W =	Width	$0 \leq W \leq 7$

The circle command will produce circles, ellipses and arcs in the graphics modes.

CIRCLE 128, 96, 6, 50 produces a red circle radius 50 centred on (128, 96)

S & F are used for arcs. S gives the start and F the finish. S must always be smaller than F as all circles are drawn from the top in a clockwise direction. S & F use the positions on a clock multiplied by 10 to give their coordinates. The 2 o'clock position is 20. The factor of 10 allows the user to pick points between the clock positions.

eg.

CIRCLE 128, 96, 6, 50, 20, 80 produces a red arc, radius 50 from the 2 o'clock position to the 8 o'clock position.

H & W

give the height width ratio for an ellipse. H/W is a fraction where  $0 \leq H \leq 7$  and  $0 \leq W \leq 7$ . This fraction can be larger or smaller than 1 by choosing suitable values.

Arcs and ellipses may be combined. Try the following demonstration program.

10 PMODE3 : CLS 7  
20 CIRCLE 50,40,6.31 : REM RED CIRCLE  
30 CIRCLE 200,150,12.3,20,80 : ~~REM~~ REM GREEN ARC  
40 CIRCLE 128,96,4,32 ; 2,5 : REM BLUE HORIZONTAL ELLIPSE  
50 CIRCLE 200,50,10,30 ; 5,2 : REM YELLOW VERTICAL ELLIPSE  
60 CIRCLE 80,150,15,30,15,85 ; 3,6 : REM WHITE ARC OF ELLIPSE  
70 GOTO 70

## Entries into Sprite Rom

(6802-2 6809)

At the start of each sprite rom there is a table of jumps (long branches 6809) to important routines. This will allow machine code programmers to have easy access to the code. The function of each routine is indicated in the table.

①	Address	Name	Function
	E002	START	Initialize code
	E005	INVEC	Get character from keyboard
	E008	OUTVEC	Print char in A to screen
	E00B	FETCH	Get char from V.P.P memory
	E00E	PLACE	Put char into V.P.P memory
	E011	SREG	Send char to V.P.P
	E014	TEXT	Set Text mode (1)
	E017	GR1	Set Graphics mode 2
	E01A	H1RES	Set hi-res (3)
	E01D	MULTI	Set Multi-colour (4)
	E020	MAP	Transfer old screen to new (update)
	E023	LOGEN	Set up V.P.P character generator
	E026	SPLPAD	Set up V.P.P sprite generator
	E029	PLUTER	Plot point in modes 3 & 4

## Important locations

V.P.P. data register	\$E000		
V.P.P. Control register	\$E001		
	Function	Value	DRAGON
XPOS	X coordinate	E6	0632 2 bit
YPOS	Y "	E7	0634 8 bit
Sprite Colour char "		E9	0636 8 bit
FRONT	Fetch memory pointer	\$F0, \$F1 (low, high)	U register 16 bit
PPOINT	Place " "	\$E0, \$EF " "	X register 16 bit

## Notes

- ① START ~~initialize~~ Sets up V.P.P and reconfigures BASIC for new handles and routines.
- ② INVEC Gets char from keyboard into A Accumulator
- ③ OUTVEC outputs character in A Accumulator to screen
- ④ FETCH FETCH Gets character into A Accumulator from V.P.P memory, location determined by FRONT
- ⑤ PLACE Puts character in A Accumulator into V.P.P memory, location determined by PPOINT.

6. SREG : sends data to U.P.P. <sup>registers.</sup> destination  
in B register (Y 6502), data in A register

7. TEXT, GRI, HIRE, MULTI : set up the for different modes

8. MAP : (not available in 6800) Transfers contents of old  
original graphics screen to new display.

9. LOVEN : Reloads character generator from ROM to U.P.P. memory  
10. SPLOAD : " Sprites " " " " " " "

11. PLOT : Plots a point onto hires or multicolor screen.  
X coord in XPOS, Y coord in YPOS, Colour in COLOUR

Full programming data on the TMS 9929A is available  
from Texas Instruments.