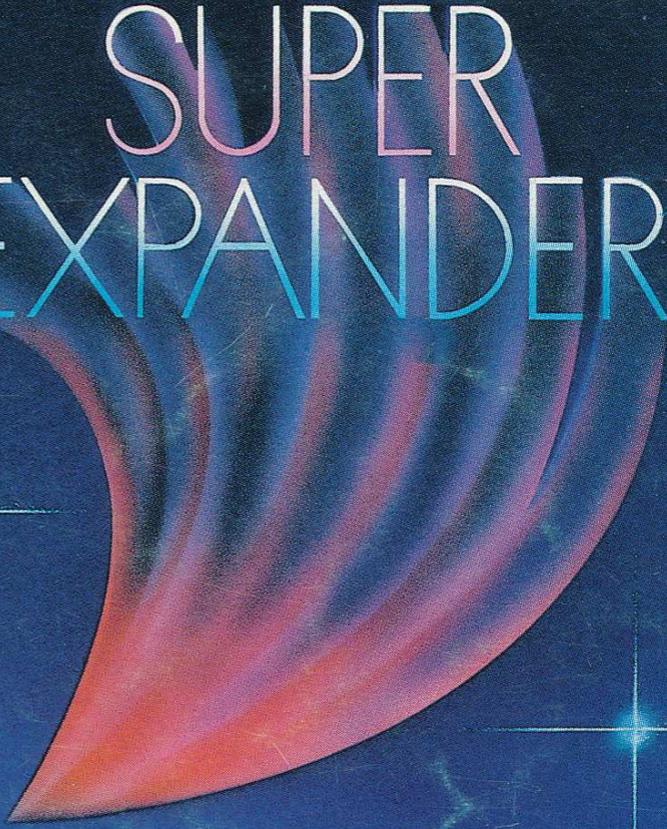


# SUPER EXPANDER



High Resolution Graphics Pack.  
Commands include: Paint, Draw,  
Circle, Colour, Point and many  
more. Includes additional 3K RAM.

# SUPER EXPANDER USER GUIDE

## TABLE OF CONTENTS

### Section One — Introduction to the Super Expander Cartridge

1.1	Introduction	1
1.2	The SUPER EXPANDER Manual	1
1.3	Loading the Cartridge	1
1.4	Super Expander Commands	2
1.5	Entering Commands	2
1.6	Conventions	2
1.7	Compatibility with the Rules of VIC BASIC	3
1.8	Storing and Running SUPER EXPANDER Programs	3
1.9	Termination of SUPER EXPANDER Programs	3
1.10	Indication of Errors	3

### Section Two — Using the Function Keys

2.1	Introduction	4
2.2	Preassigned Functions	4
2.3	Changing Key Assignments	4
2.4	Adding Carriage Returns	5

### Section Three — Plotting Graphics with Super Expander

3.1	Introduction	6
3.2	Screen Coordinates	6
3.3	Color Registers	6
3.4	Graphics Commands	6
3.4.1	GRAPHIC	7
3.4.2	COLOR	7
3.4.3	POINT	8
3.4.4	REGION	8
3.4.5	DRAW	8
3.4.6	CIRCLE	9
3.4.7	PAINT	9
3.4.8	CHAR	9
3.4.9	SCNCLR	10
3.5	Running the Example	10

### Section Four — Making Music with Super Expander

4.1	Introduction	11
4.2	Sound Registers	11
4.3	Music Commands	11
4.3.1	SOUND — Playing Chords	11
4.3.2	CTRL/left-arrow — Entering Music Mode	12
4.3.3	P — Displaying Musical Characters	12
4.3.4	Q — Cancelling Musical Character Display	12
4.3.5	V — Setting Volume	13
4.3.6	S — Choosing a Music Voice	13
4.3.7	O — Octave Selection	13
4.3.8	T — Choosing a Tempo	13
4.3.9	C,D,E,F,G,A,B — Music Notes	14
4.3.10	R — Playing Rests	14
4.3.11	# — Playing Sharps	14
4.3.12	\$ — Playing Flats	14
4.4	Summary	14

The information in this manual has been reviewed and is believed to be entirely reliable. No responsibility, however, is assumed for inaccuracies. The material in this manual is for information purposes only, and is subject to change without notice.

© 1982 COMMODORE INTERNATIONAL. All rights reserved. No part of this program or accompanying instruction leaflet may be duplicated, copied, transmitted or reproduced in any form or by any means without the prior written permission of Commodore Home Computer Division.

**Commodore Home Computer Division**  
675 Ajax Avenue, Slough Trading Estate,  
Slough, Berks. SL1 4BG England.

Printed in England

## Section Five — Commands to Read Values

5.1	Introduction .....	15
5.2	RGR .....	15
5.3	RCOLR .....	15
5.4	RDOT .....	16
5.5	RSND .....	16
5.6	RPOT .....	16
5.7	RPEN .....	16
5.8	RJOY .....	17
5.9	The Use of Read Functions .....	17

## Section Six — Summary

6.1	Introduction .....	18
6.2	The Program Description .....	18
6.3	The Program Listing .....	18

# TABLE OF FIGURES

Figure	Title	
3-1	Text Display .....	10
5-1	Joystick Values .....	17

# SECTION ONE INTRODUCTION TO THE SUPER EXPANDER CARTRIDGE

## 1.1 Introduction

The VIC 1211A SUPER EXPANDER CARTRIDGE is an interesting and versatile programming aid. It adds new colors, new sets of graphics and an enlarged music-writing potential to the operating system of your VIC 20 computer. It also contains an extra 3K of memory (RAM) to enable you to write and RUN larger programs, programmable function keys to speed program writing and extra commands to plot graphic shapes and paint colors on the screen. The extra commands added by SUPER EXPANDER will extend the programming power of your VIC.

This manual is not designed to teach BASIC programming on the VIC. If you have only a limited knowledge of BASIC programming, please refer to one of the following:

VIC 20 User Manual (supplied with your computer)

An Introduction to BASIC Parts 1 and 2 by Andrew Colin (part of the VIC learning series).

## 1.2 The SUPER EXPANDER Manual

This manual is divided into six parts as outlined below:

### SECTION ONE — INTRODUCTION TO THE SUPER EXPANDER CARTRIDGE

This section discusses the SUPER EXPANDER in broad terms as well as explaining how to load the cartridge. It shows the range of commands supplied by the cartridge, how these commands are entered and what error messages will be displayed in the event of an incorrectly entered command. The compatibility of SUPER EXPANDER commands with VIC BASIC is defined.

### SECTION TWO — USING THE FUNCTION KEYS

This section describes the commands that are automatically assigned to the four function keys on the VIC keyboard and how the programmer may assign his own commands to these keys if he wishes.

### SECTION THREE — PLOTTING GRAPHICS WITH SUPER EXPANDER

This section explains how the screen is divided for the purpose of graphics plotting and how the values indicating the colors used to draw graphics shapes are recorded. The format of each command is stated, its purpose given its use is explained and an example given in the form of a program entry. The commands are listed in the order that they might be used to write a BASIC program on the VIC such as the one contained in Section Six of this manual.

### SECTION FOUR — MAKING MUSIC WITH SUPER EXPANDER

This section describes the SUPER EXPANDER commands used to create music on the VIC. The format of each command is given, its purpose stated, its use explained and an example given in the form of a program entry. The commands are listed in the order that is required to create and play a SUPER EXPANDER musical phrase on the VIC.

### SECTION FIVE — COMMANDS TO READ VALUES

SUPER EXPANDER enables you to read the values in the registers used by the cartridge and the values returned by devices such as light pens, paddles, and joy sticks. The commands to do this are explained in this section.

### SECTION SIX — SUMMARY

This section describes a program that was written using the additional SUPER EXPANDER commands. A full program listing is also included.

## 1.3 Loading the Cartridge

Always ensure that the power supply is off before inserting or removing the SUPER EXPANDER CARTRIDGE. The cartridge is inserted into the expansion port of your VIC with the label facing up. (See your VIC 20 User Guide.) If a VIC 1010 Memory Expansion Board is in use, this also must be turned off. The SUPER EXPANDER cartridge may be used in conjunction with an 8K or 16K RAM pack. It may not be used

with a 3K RAM pack as this already forms part of the SUPER EXPANDER cartridge.

To begin using SUPER EXPANDER, simply turn the power on whilst the cartridge is in place. Your VIC will display:

★ ★ ★ ★ CBM BASIC V2 ★ ★ ★ ★  
6519 BYTES FREE  
READY

All the SUPER EXPANDER commands, the new colors and the new graphics sets are now included in the operating system of your VIC. You may use these commands at any time like any other BASIC command. You may also write longer programs using the extra 3K of memory (RAM) that is part of the cartridge.

The initialization of the cartridge automatically assigns eight SUPER EXPANDER commands to the four function keys on the right of the VIC's keyboard, e.g. DRAW, CIRCLE, PAINT. These commands are displayed on the screen when either the function key is pressed on its own, or

the  or  key is held down and a function key is pressed. (See Section 2.2.) The programmer is also given the facility to assign his own values to these keys, i.e. different from those generated by the cartridge. (See Section 2.3.)

## 1.4 SUPER EXPANDER Commands

The following is a list of SUPER EXPANDER commands which are added to the operating system of your VIC when the cartridge is inserted:

Commands used to create graphics:  
GRAPHIC, COLOR, REGION, POINT, DRAW, CIRCLE, PAINT, CHAR, SCNCLR

Commands used to write music:  
CTRL (left arrow), SOUND,  
P,Q,V,S,O,T,R,C,D,E,F,G,A,B, #, \$

Read Functions:

RGR, RCOLR, RDOT, RSND, RPOT, RPEN, RJOY

Added to the list above is the command KEY which enables the programmer to assign his own values to the function keys. (See Section 2.3.)

## 1.5 Entering Commands

All SUPER EXPANDER commands to create graphics may be used both in direct mode or as part of a program with the exception of the commands REGION and DRAW. These cannot be entered directly, i.e. they must be part of a program. In direct mode, the command is entered on the screen without a line number and is executed when the RETURN key is pressed.

Note that in direct mode characters typed following the commands GRAPHIC1, GRAPHIC2 and GRAPHIC3 will not be displayed on the screen. Extreme care must therefore be taken to ensure that any characters you type after these commands are typed correctly. Failure to do so will cause any display that may already have been on the screen to disappear. The message "SYNTAX ERROR" will then be displayed on the VIC's normal screen.

SUPER EXPANDER commands used to write music can be used in both direct and indirect modes. In direct mode, the command SOUND is typed in and then executed by pressing the RETURN key. The remaining music commands are executed in direct mode simply by pressing the key assigned to the particular command.

All SUPER EXPANDER functions except KEY can be used in your program just as you would use any other BASIC function.

The command KEY can be used in direct mode only. It is typed in and then executed by pressing the RETURN key.

## 1.6 Conventions

The format of each SUPER EXPANDER command in this manual is presented using the following notation:

1. Items written in capital letters should be typed exactly as shown. There is no need to use the SHIFT key to obtain capitals.
2. Those items written in lower case letters indicate a user-specified entry such as a pair of screen coordinates or an octave number.
3. The round brackets in the format of each SUPER EXPANDER read function must be typed in where shown.
4. Other symbols such as commas and quotation marks must be typed exactly as shown.
5. The return key is indicated by [RETURN].
6. VIC does not recognise the word "COLOUR".

The parameters used in the Super Expander commands are listed below:

Parameter	Value
n	An integer number
sc	Screen color
bo	Border color
cha	Character color
au	Auxiliary color
c	Color
cr	Color Register
x,y	Screen Coordinates
rx	Half the width of a circle or ellipse
ry	Half the height of a circle or ellipse
as	Starting point of an arc
af	Finishing point of an arc
v	Volume of sound

## 1.7 Compatibility with the Rules of VIC BASIC

The SUPER EXPANDER commands follow the same rules as VIC BASIC with one exception which is the BASIC code THEN. If this code precedes a SUPER EXPANDER command, a colon must be inserted after the code, e.g. THEN:SCNCLR. In all other respects, the SUPER EXPANDER commands are used in the same way as VIC BASIC commands.

## 1.8 Storing and Running SUPER EXPANDER Programs

A program written using the SUPER EXPANDER will not run unless the SUPER EXPANDER cartridge is in place on your VIC. The program may be saved on or loaded from diskette or cassette in the normal way.

## 1.9 Termination of SUPER EXPANDER Programs

To halt a program using SUPER EXPANDER commands whilst the program is running, hold down the RUN/STOP key and press the RESTORE key. In this way, you will return to the VIC's normal screen.

## 1.10 Indication of Errors

The message "SYNTAX ERROR" will be displayed if any SUPER EXPANDER command is incorrectly spelt. You must then type the correct spelling. This same error message will also be displayed if any parameter is omitted from a command, e.g. CIRCLE with no height specified.

# SECTION TWO

## USING THE FUNCTION KEYS

### 2.1 Introduction

The SUPER EXPANDER CARTRIDGE uses the four function keys on the VIC's keyboard to enable fast, efficient program writing. There are only four function keys but these give access to a total of eight commands.

If you press the keys normally, the functions assigned to function keys F1, F3, F5 and F7 will be displayed on the screen. Holding down either

the SHIFT or  key and pressing these same keys will display those functions assigned to function keys F2, F4, F6 and F8.

### 2.2 Preassigned Functions

If you type KEY and press RETURN, the VIC will display a list of eight functions which are automatically assigned to the function keys when the SUPER EXPANDER CARTRIDGE is initialized. These functions are listed below:

KEY 1, "GRAPHIC"  
KEY 2, "COLOR"  
KEY 3, "DRAW"  
KEY 4, "SOUND"  
KEY 5, "CIRCLE"  
KEY 6, "POINT"  
KEY 7, "PAINT"  
KEY 8, "LIST" + CHR\$(13)

### 2.3 Changing Key Assignments

FORMAT : KEY

or : KEY n, "value"

PURPOSE: To assign a new command to a function key.

KEY allows you to display the values assigned to the function keys or to assign your own value to any function key, i.e. different from that generated by the SUPER EXPANDER cartridge. If you type KEY and press RETURN, the VIC will display a list of eight functions which are automatically assigned to the function keys when the cartridge is initialized.

The value of a function key can be changed in two ways. The first of these methods is used when the function keys have been displayed as

above. To change an assignment, move the cursor to the first position following the left hand set of quotation marks of the KEY number you wish to change. Overwrite the existing command, delete any remaining characters assigned to that key and press RETURN. SUPER EXPANDER will enter the right hand set of quotation marks automatically and store the new assignment.

SUPER EXPANDER also includes a more direct method of changing the value of a single key. Simply type KEY, the number of the key you wish to change, a comma, a set of quotation marks and the value you wish to assign to that key. When you press RETURN, the second set of quotation marks will be added and the new assignment will be recorded.

One hundred and twenty-eight is the maximum number of characters that may be assigned to any one function key. You may assign this number of characters to any or all of the function keys.

EXAMPLE : To assign the BASIC code RESTORE to the F5 function key:

COMMAND: KEY [RETURN]

DISPLAY : KEY 1, "GRAPHIC"  
KEY 2, "COLOR"  
KEY 3, "DRAW"  
KEY 4, "SOUND"  
KEY 5, "CIRCLE"  
KEY 6, "POINT"  
KEY 7, "PAINT"  
KEY 8, "LIST" + CHR\$(13)

ACTION : Move the cursor to the first position following the left-hand set of quotation marks on the line displaying KEY 5.

TYPE : RESTORE [RETURN]

RESULT : The BASIC code RESTORE is now assigned to the F5 key and will be displayed on the screen each time the F5 key is pressed.

### 2.4 Adding Carriage Returns

To eliminate the need to press RETURN following a function key command, you may add a carriage return to the command in the following way:

- Change the assignment (see Section 2.3) of the key where you wish to add a carriage return, type the end quote marks but do not press RETURN.
- Type + CHR\$(13) and press RETURN.

Now, when you press this key, you will automatically generate a RETURN following the command.

EXAMPLE : To assign the BASIC code RUN and a built-in carriage return to the F7 function key:

COMMAND: KEY 7, "RUN" + CHR\$(13) [RETURN]

RESULT : You may now run a program simply by pressing the F7 key.

# SECTION THREE

## PLOTTING GRAPHICS WITH SUPER EXPANDER

### 3.1 Introduction

In Section Three, the new commands supplied by the SUPER EXPANDER cartridge to create graphics are described. The commands enable you to plot points, draw shapes, enter text and paint on the screen in any one of fifteen colors without having to access locations in memory. These commands are entered just like any other BASIC command and may be used in direct and indirect mode. The way the VIC holds the colors you have selected in color registers is also explained. The commands are listed in the order in which they might be used to write a program such as the one in Section Six of this manual.

### 3.2 Screen Coordinates

For the purpose of plotting graphics the screen is divided into a 1024 by 1024 matrix. A matrix is simply a grid in which a particular location is addressed by referring to its coordinates. A coordinate is a number used to state the distance of a point on a grid from the origin or 0 point. Two numbers are required, one indicating how far the point is away from the origin in a horizontal direction and the other indicating how far the point is from the origin in a vertical direction. The horizontal is called the x axis and the vertical is called the y axis. The first coordinate stated in all plotting commands is the x coordinate. In SUPER EXPANDER, this is the position of a point on the screen relative to the left edge of the screen. The second coordinate is the y coordinate. This indicates how far down from the top of the screen the point is located. The origin or zero point of the screen grid in SUPER EXPANDER is the upper left corner of the screen. The coordinates of any point on the screen are in the range 0 to 1023.

### 3.3 Color Registers

When the VIC is turned on with the SUPER EXPANDER cartridge in place, four color registers are assigned. (A register is simply a special memory location used to hold a value.) A number in the range 0 to 15 is stored in each register and is used to indicate a color. The

colors and their associated values are listed below:

- 0 Black
- 1 White
- 2 Red
- 3 Cyan
- 4 Purple
- 5 Green
- 6 Blue
- 7 Yellow
- 8 Orange
- 9 Light Orange
- 10 Pink
- 11 Light Cyan
- 12 Light Purple
- 13 Light Green
- 14 Light Blue
- 15 Light Yellow

Register 0 holds the value of the color used as the screen background in a graphics display.

Register 1 contains the value of the color used to form the border surrounding the graphics screen.

Register 2 holds the value of the character color, i.e. the color for plotting points, drawing graphic shapes or displaying text on a graphics screen.

Register 3 holds the value of the auxiliary color which also may be used to plot graphic shapes. This register may only be accessed in multi-color mode or mixed mode. (See Section 3.4.1.)

Initial color values are placed in these registers by executing the command COLOR. (See Section 3.4.2.) When the system is turned on, the values in the registers are 1, 3, 6, and 0 respectively.

### 3.4 Graphics Commands

All examples for commands used to create graphics are given in the following sections in the form of program entries. You may build up the program and RUN it after each new line is input. If you do so, before inputting the next line you must hold down the RUN/STOP key and press the RESTORE key.

#### 3.4.1 GRAPHIC

FORMAT : GRAPHIC n

PURPOSE: To select a graphics set from the VIC's operating system.

The command GRAPHIC may be used in direct and indirect mode. There are four different GRAPHIC modes, each of which selects a set of graphics from the VIC's operating system.

GRAPHIC 0 is the VIC's normal mode. This command is used to return to a normal screen display from a graphics screen. It is most commonly used at the end of a program containing graphics.

GRAPHIC 1 selects the multi-color graphics set from the VIC's operating system. In this mode, the VIC scales down the graphics plotting so that each position on the screen occupies an area 16 dots wide and 8 dots deep within the 1024 by 1024 dot screen matrix. (See Section 3.2.) Points may be plotted on the screen using any one of the four color registers available. (See Section 3.3.) All 15 SUPER EXPANDER colors may be used in this mode.

GRAPHIC 2 activates the high-resolution color graphics set in the VIC's operating system. In this mode, each position on the screen occupies an area 8 dots wide and 8 dots deep within the screen matrix. Thus this mode displays points on the screen in a resolution twice as fine as that of the GRAPHIC 1 command. However, in this mode the colors in which points may be drawn on the screen are restricted to those contained in registers 0 and 2, i.e. the screen or character colors. Only those colors numbered 0 to 7 may be used in this mode. (See Section 3.3.) If you specify a color value greater than 7 in the COLOR command, SUPER EXPANDER will subtract 8 from that value and use the color corresponding to the calculated value instead. The character color may be changed using the command REGION if you wish to use more than one color to plot points on the screen. (See Section 3.4.4.)

GRAPHIC 3 activates either the multi-color or high-resolution graphics set depending on the value contained in the character color register after the command COLOR has been executed. (See Section 3.4.2.) If the value of the color lies within the range 0 to 7, all plotting will take place in high-resolution mode. If the color number is larger than seven, all points will be drawn on the screen in multi-color mode. This command is useful if you wish to use high-resolution and multi-color graphics on the same screen.

EXAMPLE : To select the high-resolution color graphics set:

PROGRAM

ENTRY: 100 GRAPHIC 2

RESULT : The high-resolution graphics set within the VIC operating system is activated.

#### 3.4.2 COLOR

FORMAT : COLOR sc,bo,cha,au

PURPOSE: To assign colors to each of the four color registers.

COLOR is used to select colors for the screen, the border, the character and the auxiliary color. The screen color, along with setting the background color, may also be used to erase points from the screen that have been plotted in a different color. The auxiliary color is used mostly with the command PAINT. (See Section 3.4.7.) The value of each color is stored in one of four color registers in VIC's memory. Once the registers have been set up with the COLOR command, the number of the color register forms the first parameter of all SUPER EXPANDER commands used to plot graphics on the screen. (See Section 3.3.)

When the system is turned on, the four color registers contain the values 1, 3, 6, and 0 respectively.

The color registers available for use in multi-color and high-resolution modes are outlined below:

<i>multi-color</i>	<i>high-resolution</i>
0 screen color	0 screen color
1 border color	1 not available
2 character color	2 character color
3 auxiliary color	3 not available

As you can see from the list above, the only color registers available for use in high-resolution mode are those containing the screen and character colors. This limitation can be eased by changing the character color with the command REGION if you require a different character color on the screen. (See Section 3.4.4.)

EXAMPLE : To select a white screen, a blue border, a black character color and a pink auxiliary color:

PROGRAM

ENTRY: 110 COLOR 1,6,0,10

RESULT : When this program line is executed, a white screen surrounded by a blue border is displayed. All graphics plotted using the color register containing the character color will be in black. If the auxiliary color is used, the plotting will be in pink.

### 3.4.3 POINT

FORMAT : POINT cr,x,y

or: POINT cr,x,y,x1,y1 . . . . .xn,yn

PURPOSE: To plot a point or points on the screen in a specified color.

In multi-color mode, a point may be plotted on the screen in any of the colors available in the four color registers. (See Section 3.3.) In high-resolution mode, points may be plotted only with those colors available in the screen or character color registers. (See Section 3.4.2.) The command may be used to plot one or a number of points on the screen. If only one set of screen coordinates is specified in the command, then only one point will be plotted. Alternatively, you may specify a number of sets of screen coordinates which will result in that number of points being plotted on the screen. You may, however, only plot in one color in a single command.

EXAMPLE : To plot a single point on the screen in the color black:

PROGRAM

ENTRY: 120 POINT2,900,900

RESULT : When this line is executed, a single black point will be displayed in the bottom right-hand corner of the screen.

EXAMPLE : To plot four points on the screen in the same color:

PROGRAM

ENTRY: 130 POINT2,900,100,900,300,1000,100,1000,300

RESULT : When this line is executed, four black points will be plotted in the top right-hand corner of the VIC's screen.

### 3.4.4 REGION

FORMAT : REGION c

PURPOSE: To change the character color

REGION changes the character color which you previously specified in the command COLOR, i.e. it changes the value stored in color register two. (See Section 3.4.2.) Any one of the 15 SUPER EXPANDER colors may be used as the parameter of this command in multi-color mode. In high-resolution mode only colors numbered 0 to 7 may be used. This command is especially valuable if you are in high-resolution mode and wish to create displays other than in the screen or character colors. (See Section 3.4.2.)

EXAMPLE : To change the character color to red:

PROGRAM

ENTRY: 140 REGION 2

RESULT : When this program line is executed, the color value for red is stored in register two. Following the command, all points drawn on the screen which use the character color will be drawn in red. This will continue until the character color is changed with another REGION command.

### 3.4.5 DRAW

FORMAT : DRAW cr,x,y TO x1,y1

or : DRAW cr,x,y TO x1,y1 TO x2,y2 . . . .

or : DRAW cr, TO x,y

PURPOSE: To plot a straight line between two points on the screen.

DRAW will plot a straight line in the color indicated in the color register specified. (See Section 3.4.2.) The DRAW command may be used in three ways:

1. You may DRAW one line between two points on the screen specifying only one set of coordinates. Then, if you wish, you may use the command again to DRAW another line between different points on the screen.
2. You may DRAW many lines with one DRAW command by specifying many sets of coordinates separated by the code TO. The first set of coordinates indicates the starting point of the first line to be drawn on the screen. The termination point of each line becomes the starting point of the next line.
3. You may also DRAW specifying only the termination point of a line. In this case, the starting point of the line will be the termination point of the last graphic shape, circle or point.

NOTE: In the latter case, if no previous shape has been drawn on the screen, the starting point of the line will be screen location 0,0, i.e. the top left-hand corner of the screen.

EXAMPLE : To draw a rectangle:

PROGRAM

ENTRY: 150 DRAW 2, 100, 100 TO 200, 100 TO 200, 200 TO 100, 200 TO 100, 100

RESULT : A small red rectangle will be drawn in the top left corner of the screen when this program line is executed.

EXAMPLE : To draw a diagonal line to the centre of the screen:

PROGRAM

ENTRY: 160 DRAW 2 TO 512, 512

RESULT : A line will be drawn from the point where VIC finished drawing the rectangle to the middle of the screen.

### 3.4.6 CIRCLE

FORMAT : CIRCLE cr,x,y,rx,ry

or : CIRCLE cr,x,y,rx,ry,as,af

PURPOSE: To draw a circular shape on the screen.

The coordinates defined in the command CIRCLE specify the screen location of the centre of a circle or circular shape. The next set of parameters give the width and height of that shape from its centre. This last set of coordinates enables you to draw flattened circles, i.e. ellipses. Because of the way points are plotted on the screen, in order to draw a precise circle, rx does not equal ry as you would expect. You must multiply the rx parameter by 0.7 to allow for the fact that the screen is rectangular although the x and y axes each have the same number of divisions.

To draw an arc of a circle you must add a further set of parameters to the command CIRCLE. The first parameter is the starting point of the arc on the circumference of the circle and the second parameter is the finishing point of the arc. The unit used in these parameters is a "gradian". There are 1000 gradians in a complete circle. The position of gradian 0 is at the 3 o'clock position on a clock face. Gradians increment in a clock-wise direction around the circumference of the circle finishing at gradian 1000, again at the 3 o'clock position.

EXAMPLE : To draw a circle:

PROGRAM

ENTRY: 165 CIRCLE2,512,512,70,100

RESULT : When this program line is executed, a red circle will be drawn in the middle of the VIC's screen.

EXAMPLE : To draw two arcs of a circle:

PROGRAM

ENTRY: 170 CIRCLE 2, 800, 300, 140, 100, 40, 60  
180 CIRCLE 2, 800, 300, 140, 100, 90, 10

RESULT : When these lines are executed, a set of round brackets will be drawn in red on the VIC's screen.

### 3.4.7 PAINT

FORMAT : PAINT cr,x,y

PURPOSE: To fill an enclosed area with color.

PAINT fills in a whole area with the color indicated in the color register defined. The area must be completely enclosed or painting will take place over the whole screen. The area to be painted is specified by the coordinates of ANY point within its boundaries.

Each area may only be painted once. If you are using the GRAPHIC 3 instruction set (see Section 3.4.1), an area which is bounded by drawing in multi-color mode must be painted in the same mode, i.e. you cannot change modes between creating the shape and painting it.

EXAMPLE : To draw a red circle and paint it in yellow:

PROGRAM

ENTRY: 190 CIRCLE 2,300,800,70,100  
195 REGION 7  
200 PAINT 2, 300,800

RESULT : When this section of the program is executed, a small circle will be drawn in red in the bottom left-hand corner of the screen and the area inside the circle will be painted in yellow.

### 3.4.8 CHAR—TEXT DISPLAY

FORMAT : CHAR row, column, "text"

PURPOSE: To display normal text on a graphics screen.

CHAR will display normal text on a graphics screen beginning at the row and column defined. This command can be used in the GRAPHIC 2 high-resolution mode and in the GRAPHIC 3 mixed mode when the value of the color contained in the character color register is less than 8. (See Section 3.4.1.) This command is not available in GRAPHIC 1 multi-color mode.

EXAMPLE : To display the word "ART" in black on the graphics screen:

PROGRAM

ENTRY: 210 REGION 0  
220 CHAR 5,14,"ART"

RESULT : When this section of the program is executed, the word "ART" will appear in black between the previously displayed set of brackets. (See Figure 3-1.)

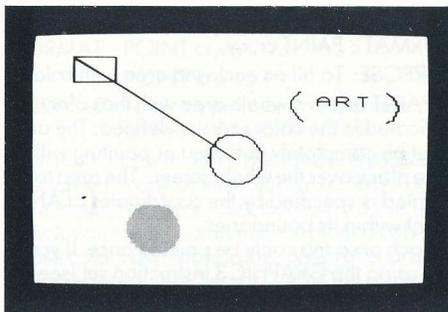


Figure 3-1 Text Display

### 3.4.9 SCNCLR

FORMAT : SCNCLR

PURPOSE: To clear a graphics screen.

SCNCLR is used in graphics mode to clear the screen in the same way that the BASIC statement PRINT "[SHIFT/CLR/HOME]" is used. This command can be used in both direct and indirect modes.

EXAMPLE : To clear the graphics screen in the program listed above:

```
PROGRAM
ENTRY: 240 SCNCLR
```

RESULT : When this line is executed, the screen will be blank, i.e. a white screen with a blue border.

### 3.5 Running the Example

To RUN the above as a complete program, add a delay loop before the screen clear and an end to the program as follows:

```
PROGRAM
ENTRY: 230 FOR X = 1 TO 2500:NEXT X
      250 GRAPHIC 0:END
```

When you run this program you will see an example of some of the simple graphics capabilities of SUPER EXPANDER.

## SECTION FOUR

# MAKING MUSIC WITH SUPER EXPANDER

### 4.1 Introduction

This section includes all the steps required to make music or noise on the VIC with SUPER EXPANDER. Both music and noise can be played in direct or indirect mode. You may play a series of notes at the same time, i.e. a musical chord, or play a string of single notes or sounds. Within a program, a music phrase using single notes is created by entering the SUPER EXPANDER music characters inside a BASIC PRINT statement and "printing" the music. Music can also be created in direct mode without the use of a PRINT statement simply by pressing the VIC's keys after entering music mode. To create a chord, the command SOUND is used in either direct or indirect mode. SOUND is also used to generate noise.

NOTE: The pitch of notes or noise may differ slightly between different VICs.

### 4.2 Sound Registers

When the VIC is turned on with the SUPER EXPANDER cartridge installed, five sound registers are assigned. A register is a special location in memory which holds a value. The values in the five sound registers are placed there after the execution of the command SOUND. (See Section 4.3.1.)

The registers are numbered 1 to 5. Registers 1 to 4 are linked to the four voices of the VIC whilst register 5 is concerned with the intensity of the sound. A value in the range 128 to 255 placed in one of the first four registers will cause a sound to be played in the associated voice. If the value is less than 128, no sound will be played through the voice.

Registers 1 to 3 will generate musical notes. (For a complete list of music notes and their associated numeric values, see Appendix F of your VIC 20 Computer User Guide.) Each musical sound register, i.e. one, two and three, has three complete octaves in its range. You can think of the first three music registers as "voices" much like the parts in singing, i.e. bass, treble and alto. Each register is one octave higher than

its predecessor, i.e. a note played in octave 3 of sound voice 1 has the same pitch as the identical note played in octave 2 of sound voice 2 and octave 1 of sound voice 3.

Register 4 is used to create "white noise" for special sound effects.

Register 5 contains the value of the volume-level of music and/or noise. This value can be between 0 (off) to 15 (loudest). (Note that in single note music mode, i.e. with the command V, you are limited to a maximum value of 9.)

### 4.3 Music Commands

In this section, the format of each SUPER EXPANDER music command is shown, its purpose given, its use described and an example given. All commands can be entered in direct or indirect mode. The command SOUND, which can create a musical "chord", is shown as a direct mode entry. An example of all other music commands is given in the form of a program entry. Each of these commands is entered on one program line. In this way, a music phrase is built step-by-step until the tune is complete and ready to play. The new entry on the line is indicated in bold type.

#### 4.3.1 SOUND — PLAYING CHORDS

FORMAT : SOUND s1, s2, s3, s4, s5

PURPOSE: To create multi-voice sound effects.

The command SOUND enables music chords to be played on the VIC. The parameters s1 to s3 define the values of musical notes each in the range 128 to 255 which are to be played in one of the three VIC musical voices. A value less than 128 for any one of these parameters will play no sound from the associated voice. (See Appendix F of your VIC User Guide.) The parameter s4 defines the value of a "white noise" used to create special sound effects. It can contain a noise value within the same range, i.e. 128 to 255, and may be switched off in the same way as voices 1 to 3. The parameter s5 defines the value of volume level set ranging from 0 (off) to 15 (loudest). The SOUND command is cancelled by assigning the value 0 to all the voices or by assigning the value 0 to parameter s5.

EXAMPLE : To play a series of linked chords:  
PROGRAM  
ENTRY: 300 FOR X = 1 TO 50: SOUND  
225,225,225,0,9  
310 NEXT X  
320 FOR X = 1 TO 50: SOUND  
230,230,230,0,9  
330 NEXT X  
340 FOR X = 1 TO 50: SOUND  
235,235,235,0,9  
350 NEXT X  
360 SOUND 0,0,0,0,0  
RESULT : A three chord triplet will be played when this section of the program is executed.

#### 4.3.2 — ENTERING MUSIC MODE

FORMAT : CTRL left arrow  
PURPOSE: To enter the SUPER EXPANDER music mode.

CTRL left arrow can be used in both direct and indirect modes to enable you to make SUPER EXPANDER music on the VIC. The command is executed by holding down the CTRL key and pressing the left-arrow key. You may use the command inside quotation marks to "print" a series of musical notes and play the tune that has been composed.

When you enter this command inside a PRINT statement, a reverse-field F will be displayed. In direct mode, you may use this command outside quotation marks. No symbol will be displayed on the screen. You may then use the keys of the VIC to play single note music just like any keyboard instrument. To terminate the music mode, press the RETURN key.

When the CTRL left-arrow command has been executed, the SUPER EXPANDER music package interprets only those characters that form the music character set. They are:

P,Q,V,S,O,T,R,C,D,E,F,G,A,B, #, \$

All other VIC keyboard characters are ignored by the SUPER EXPANDER cartridge.

Within your program, you may enter up to 72 of these characters in any one PRINT statement between the CTRL left arrow symbol and the last set of quotation marks. By reducing this figure to 71 and adding a semi-colon following the last set of quotation marks, the next string of musical notes will be played immediately following the previous string of notes. To stop a tune while it is playing, simply hold down the RUN/STOP key and press the RESTORE key.

EXAMPLE : To write music in program mode:

PROGRAM  
ENTRY: 370 PRINT "(CTRL/left-arrow)"

RESULT : A reverse-field F will be displayed. Any key you type between this character and the last set of quotation marks, until RETURN is pressed, will be interpreted by the SUPER EXPANDER music package when the program is run.

EXAMPLE : To begin making music in direct mode:

ACTION : Hold down the CTRL key and press the left arrow key.

RESULT : No character will be displayed on the screen but all music character keys will be interpreted by the SUPER EXPANDER music package. Any key pressed that represents a music note, e.g. C,D or E will result in that note being played.

#### 4.3.3 P — DISPLAYING MUSICAL CHARACTERS

FORMAT : P  
PURPOSE: To display all music characters on the screen.

The command P can be used in direct and indirect mode. It is used to display the characters that you have used to create a piece of music as that music is being played. This command is especially useful as a composing aid. It will help you to see where any wrong notes appear in your music composition. You may then correct these notes with the VIC's normal editing facilities. The command Q is used to halt music character display. (See Section 4.3.4.)

EXAMPLE : To display music characters in program mode:

PROGRAM  
ENTRY: 370 PRINT "(CTRL/left-arrow) P"  
RESULT : When this line is executed, all music characters following the P on the line will be displayed on the screen.

#### 4.3.4 Q — CANCELLING MUSICAL CHARACTER DISPLAY

FORMAT : Q  
PURPOSE: To cancel the display of musical characters.

The command Q can be used both in direct and indirect mode. The command will halt the display of characters during music mode.

EXAMPLE : To play music characters in direct mode with no screen display:

ACTION : Hold down the CTRL key and press the left-arrow key to enter the music mode. Type P and press keys C and D.

RESULT : Two notes are played and the characters C and D are displayed on the screen.

ACTION : Type Q and press the C and D keys again.

RESULT : The same two notes are played but no characters are displayed on the screen.

#### 4.3.5 V — SETTING VOLUME

FORMAT : V n  
PURPOSE: To set the music volume.

The command V sets the volume which will be used to play single notes of SUPER EXPANDER music, i.e. it is not used in conjunction with the command SOUND. This volume value can range from 0 (off) to 9 (loudest).

EXAMPLE : To set a volume of 9 prior to music writing in program mode:

PROGRAM  
ENTRY: 370 PRINT "(CTRL/left-arrow) PV9"

RESULT : When the line is executed, all music between the command and the second set of quotation marks will be played with the volume set at the highest level. This level of volume will continue until the command is used again to set a quieter volume.

#### 4.3.6 S — CHOOSING A MUSIC VOICE

FORMAT : S n  
PURPOSE: To select a music voice.

The command S is used to select one of the four sound voices available in VIC music mode. The voice chosen determines the "pitch" of the note or how high the note will be played. Each voice has three octaves in its range. (See Section 4.3.7.) An octave is a group of eight notes. The voices are numbered 1 to 4. Voices 1 to 3 are used to compose music with SUPER EXPANDER. These voices are analogous with Bass, Treble and Alto singing voices respectively. Voice 4 is used for creating "white noise" sound effects.

EXAMPLE : To select sound voice 2:

PROGRAM  
ENTRY: 370 PRINT "(CTRL/left-arrow) PV9S2"

RESULT : All subsequent music will be played using the second sound voice until the command is changed.

#### 4.3.7 O — OCTAVE SELECTION

FORMAT : O n  
PURPOSE: To select a music octave.

The command O (the letter O as in Oscar not the number 0) is used to select a group of eight notes from which you wish to play a note. Each musical sound register, i.e. one, two and three, has three complete octaves in its range. Each sound voice begins one octave higher than its predecessor, i.e. a note played in octave 3 of sound voice 1 has the same pitch as the identical note played in octave 2 of sound voice 2 and octave 1 of sound voice 3. When playing notes in different sound voices, always ensure that the octaves are chosen correctly, i.e. to sing in close harmony, parts or unison.

EXAMPLE : To select octave 3 in the preceding program line:

PROGRAM  
ENTRY: 370 PRINT "(CTRL/left-arrow) PV9S2O3"

RESULT : When this line is executed, all notes between the 3 and the second set of quotation marks will be played in the third octave of the second sound voice.

#### 4.3.8 T — CHOOSING A TEMPO

FORMAT : T n  
PURPOSE: To select the duration of a note.

The command T selects the duration or tempo of a note according to the values in the following chart:

TEMPO No.	BEATS/ MINUTE	DURATION (★ 1/60 sec)
0	900	4
1	600	6
2	450	8
3	300	12
4	225	16
5	150	24
6	112.5	32
7	56.25	64
8	28	128
9	14	255

EXAMPLE : To select a tempo of 112.5 beats/minute in the preceding program line:

PROGRAM  
ENTRY: 370 PRINT "(CTRL/left-arrow) PV9S2O3T6"

RESULT : The note(s) between T6 and the second set of quotation marks will be played for a duration of 32/60ths of a second.

### 4.3.9 C,D,E,F,G,A,B — MUSIC NOTES

FORMAT : letter

where letter is any one or combination of the characters CDEFGAB.

PURPOSE: To play a musical note(s).

The keys C,D,E,F,G,A and B are used to play notes in either direct or indirect mode. In direct mode, the note is played as the key is pressed. In indirect mode, the notes are enclosed in quotation marks as part of a PRINT statement. There is no need to place separating characters between musical notation in either mode. The SUPER EXPANDER music package will ignore any non-music character.

EXAMPLE : To play a melody in the preceding program line:

PROGRAM  
ENTRY: 370 PRINT“(CTRL/left-arrow)  
PV9S2O3T6**BCDE**”

RESULT : When this line is executed, three notes will be played and the musical characters will be displayed on the screen.

### 4.3.10 R — PLAYING RESTS

FORMAT : R

PURPOSE: To play a musical rest.

R is used to create silent periods in a musical composition. The length of the period, or rest, is determined by the command T. (See Section 4.3.8.)

EXAMPLE : To insert rests in the preceding program line:

PROGRAM  
ENTRY: 370 PRINT“(CTRL/left-arrow)  
PV9S2O3T6**CRDRE**”

RESULT : When this line is now executed, the notes will be separated by rest beats.

### 4.3.11 # — PLAYING SHARPS

FORMAT : #

PURPOSE: To indicate that the following note is sharp.

The command # is used in music mode to instruct the SUPER EXPANDER music package to play the next note sharp. Only the note immediately following the # is affected.

EXAMPLE : To play the notes C and C # in direct mode:

ACTION : Type PRINT“(CTRL/left-arrow)  
PV9S2O3T6**CR # C**”[RETURN]

RESULT : Two notes will be played, the second a semi-tone higher than the first.

### 4.3.12 \$ — PLAYING FLATS

FORMAT : \$

PURPOSE: To play the next note flat.

The command \$ is used in the music mode to instruct the music package to play the following note flat. Only the note following the command will be affected.

EXAMPLE : To play the notes B and B\$ in direct mode:

ACTION : Type PRINT“(CTRL/left-arrow)  
PV9S2O3T6**BR\$B**”[RETURN]

RESULT : Two notes will be played, the second a semi-tone lower than the first.

## 4.4 Summary

Putting together all the music commands listed above, you can play a simple tune such as the one below:

PROGRAM  
ENTRY: 370 PRINT“(CTRL/left-arrow)  
PV9S2O3T6**CDET4CRT  
6CDET4CRT6EFT7GT  
4RT6EFT7G**”

# SECTION FIVE

## COMMANDS TO READ VALUES

### 5.1 Introduction

This section illustrates the seven read functions that are part of the SUPER EXPANDER command set. These functions can be used in direct and indirect mode. In indirect mode, the function must be preceded by the BASIC code PRINT and the RETURN key pressed following the function to display the value read. The functions enable you to determine values created by certain SUPER EXPANDER commands or by games control devices. For example, using one of these functions, you can read the values of the color registers you set up with the command COLOR or the sound registers you set up with the command SOUND. If you incorporate the use of games paddles, the joystick or the light pen in your program, you can use a function to determine the value indicated by a games device at a moment in time. The format of each function is shown, its purpose given, its use explained and an example shown.

NOTE: In the formats listed below, the brackets must be included in the command statement.

### 5.2 RGR

FORMAT : RGR (n)

PURPOSE: To read the graphics mode that was set with the command GRAPHIC.

This function reads the number of the graphics mode that was set with the command GRAPHIC. (See Section 3.4.1.) Any number in the range 0 to 255 can be used as the parameter of this function. Whilst a parameter must be stated, its value does not affect the value returned by the function. This function is useful in program mode when you wish to change from one graphics mode to another and then return to the original graphics mode. This function is not valuable in direct mode because it always returns a value of 0.

EXAMPLE : To draw three circles, changing color and graphics mode before each:

PROGRAM  
ENTRY: 10 GRAPHIC 2  
20 COLOR 1,6,0,10

30 CIRCLE 2,512,512,70,100  
40 X = RGR (0)  
50 GRAPHIC 3  
60 REGION 10  
70 CIRCLE 3,850,750,70,100  
80 GRAPHIC X  
90 REGION 0  
100 CIRCLE 2,300,800,70,100  
110 FOR Z = 1 TO 1000: NEXT:  
GRAPHIC 0

RESULT : When the above program is RUN, the number of the first graphics mode is assigned to the variable X and that variable is used to set up the third graphics mode.

### 5.3 RCOLR

FORMAT : RCOLR (cr)

PURPOSE: To read the color contained in a color register.

This function will read the value of the color contained in the color registers which you assigned by the command COLOR or changed with the command REGION. The function is especially useful in high-resolution mode when you wish to plot a point on the screen with a color other than the screen and character color and do not wish to use the command REGION.

EXAMPLE : To draw a circle in high-resolution using the character color and paint it in the auxiliary color:

PROGRAM  
ENTRY: 10 GRAPHIC 2  
20 COLOR 1,6,0,2  
30 CIRCLE 2,512,512,70,100  
40 Q = RCOLR (3)  
50 REGION Q  
60 PAINT 2, 512,512

RESULT : When the above program is RUN, the color contained in color register 3 will be read and that color will be used to paint the circle.

## 5.4 RDOT

FORMAT : RDOT (x,y)

PURPOSE: To obtain the color value of a dot on the screen.

The RDOT function will read the color value of a dot located at the position in the screen matrix given in the function parameters. (See Section 1.4.) This function is most useful in multi-color mode when you wish to plot points or draw graphics shapes on the screen with a paint color.

EXAMPLE : To draw and paint a circle in multi-color mode and draw a second circle with the color used when painting:

PROGRAM

```
ENTRY: 10 GRAPHIC 1
      20 COLOR, 1,6,0,10
      30 CIRCLE 2,512,512,70,100
      40 PAINT 3,512,512
      50 X= RDOT(512,512)
      60 REGION X: CIRCLE 2,750,
          750,70,100
```

RESULT : When this program is RUN, a black circle painted in pink and a pink circle will be displayed on the screen.

## 5.5 RSND

FORMAT : RSND (n)

PURPOSE: To read the value of a sound register.

The function RSND reads the value of a note in the first four sound registers set up by SUPER EXPANDER and the value of the volume set in the fifth sound register. (See Section 4.2.) This function can only be used to determine the contents of the sound registers following the command SOUND. (See Section 4.3.4.)

EXAMPLE : To use the RSND function within a program:

ACTION : Type in the following short program and RUN it:

PROGRAM

```
ENTRY: 10 X = 225 : SOUND X,X,X,0,9
      20 Y = RSND (5)
      30 FOR Z = Y TO 0 STEP -1
      40 SOUND X,X,X,0,Z
      50 FOR C = 1 TO 500 : NEXT C,Z
      60 FOR Z = 0 TO Y
      70 SOUND X,X,X,0,Z
      80 FOR C = 1 TO 500 NEXT C,Z
      90 SOUND X,X,X,0,0
```

RESULT : A musical chord will be played, first falling and then rising in volume.

## 5.6 RPOT

FORMAT : RPOT (n)

PURPOSE: To read the value of a paddle.

The function RPOT will read the value of a paddle and return a figure in the range 0 to 255. This represents the position of the paddle relative to the left edge of the screen. The value of n is 0 to read one paddle and 1 to read the other.

EXAMPLE : To read the value of a paddle in direct mode:

ACTION : Connect a paddle set to the game port on your VIC. (See your VIC 20 User Guide and accompanying literature.) Turn the paddle control in a clockwise direction until you reach the furthest position.

TYPE : PRINT RPOT (0) [RETURN]

DISPLAY : 0

ACTION : Turn the paddle control as far as it can go in the opposite direction.

TYPE : PRINT RPOT (0) [RETURN]

DISPLAY : 255

## 5.7 RPEN

FORMAT : RPEN (n)

PURPOSE: To read the value of the light pen.

The function RPEN will read the position on the screen at which the light pen is pointing. If n = 0 the position relative to the left edge of the screen will be read (the X value) and if n = 1 the position of the pen in relation to the top of the screen will be read (the Y value). A value in the range 0 to 255 will be returned in both cases.

NOTE: If the values returned by this function fall outside this range, your light pen will need to be calibrated. To do this, refer to the instructions supplied with your pen.

EXAMPLE : To read the point on the screen at which the light pen is pointing:

ACTION : Connect a light pen to the games port on your VIC.

Type in the following program and RUN it:

PROGRAM

```
ENTRY: 10 X = RPEN(0):Y = RPEN(1):
      PRINT X,Y:GOTO 10
```

ACTION : Point the light pen at different positions on the screen.

RESULT : A series of pairs of numbers will scroll up the screen. The number on the left is the X value and the number on the right is the Y value.

## 5.8 RJOY

FORMAT : RJOY (n)

PURPOSE: To read the value of the joystick.

The function RJOY determines the value which indicates a joystick movement. You may use any number between 0 and 255 as the parameter of this function. Whilst a parameter must be stated, its value does not affect the value returned by the function.

The values generated by the joystick when it is moved in each direction and when the fire-button is pressed is given on the chart below. A diagram showing these values is Figure 5-1.

DIRECTION	VALUE
UP	1
DOWN	2
RIGHT	8
LEFT	4
RIGHT DIAGONAL UP	9
LEFT DIAGONAL UP	5
RIGHT DIAGONAL DOWN	10
LEFT DIAGONAL DOWN	6
FIRE BUTTON	128
NO MOVEMENT	0

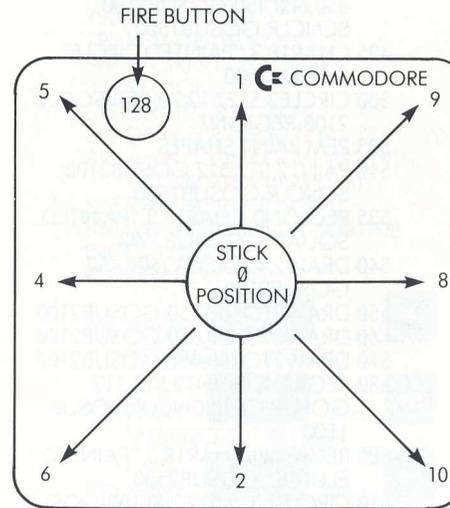


Figure 5-1. Joystick Values

EXAMPLE : To return all possible values of the joystick:

ACTION : Insert the joystick into the games port on your VIC. (See your VIC User Guide.)

Type in and then RUN the following short program:

PROGRAM

```
ENTRY: 10 X = RJOY (0) : PRINT X
      20 FOR S = 1 TO 1000 : NEXT
      30 GOTO 10
```

ACTION : Move the joystick around and press the fire-button whilst the program is running.

RESULT : The values of all the joystick positions and of the fire-button are displayed on the screen.

## 5.9 The Use of Read Functions

The functions which read the values of games devices have an obvious use. When you write games, you will need to know the position that the player has indicated. RPOT, RJOY and RPEN enable you to do this with ease.

The chief value of the other commands is as a debugging or diagnostic tool. If your program is not running correctly, you may, for example, read the values of the color or sound registers at a particular point and see if the values returned are what you expected.

# SECTION SIX

## SUMMARY

### 6.1 Introduction

This section contains a program that was written using the SUPER EXPANDER cartridge. The program is described and a full program listing is given in order that you may try the program out yourself.

### 6.2 Program Description

The first part of the program contained in Section 6.3 illustrates some of the many shapes that can be plotted on the screen using the SUPER EXPANDER high-resolution graphics set. The second part of the program incorporates music into the program while the corresponding notes are actually being drawn on a music stave on the screen.

### 6.3 Program Listing

The following is a complete listing of the program:

```

6 REM ALL CYAN SCREEN
8 POKE36879,59
10 GOSUB1200

12 PRINT"  SHIFT  AND  CLR HOME
  CSRS CSRS CSRS CSRS
7 spaces ARTWORK":GOSUB1300
16 FORG=1TO100:NEXT
17 REM SELECT GRAPHICS
18 GRAPHIC2
19 REM SELECT COLORS
20 COLOR3,3,0,10:GOSUB1500:
  GOSUB2100
30 REM DRAW SQUARE
35 CHAR18,7,"SQUARE":GOSUB
  2500
40 DRAW2,400,450TO600,450:
  GOSUB2000
60 DRAW2TO600,650:GOSUB2000
80 DRAW2TO400,650:GOSUB2000
100 DRAW2TO400,450:GOSUB2000:
  SCNCLR:GOSUB1500
130 REM DRAW CIRCLE
135 CHAR18,7,"CIRCLE":GOSUB2500
140 CIRCLE2,512,512,70,100:GOSUB
  2100:SCNCLR:GOSUB1500
  
```

```

170 REM DRAW TRIANGLE
175 CHAR18,7,"TRIANGLE":
  GOSUB2500
180 DRAW2,550,450TO350,650:
  GOSUB2000
200 DRAW2TO750,650:GOSUB2000
220 DRAW2TO550,450:GOSUB2000:
  SCNCLR:GOSUB1500
250 REM DRAW ELLIPSE
255 CHAR18,8,"ELLIPSE":GOSUB2500
260 CIRCLE2,512,512,200,100:GOSUB
  2100:SCNCLR:GOSUB1500
300 REM DRAW ARC
305 CHAR18,9,"ARC":GOSUB2500
310 CIRCLE2,512,512,70,100,25,75:
  GOSUB2100:SCNCLR:GOSUB1500
350 REM PLOT POINTS
355 CHAR18,8,"POINTS":GOSUB2500
360 POINT2,400,450,400,650,600,
  650,600,450:GOSUB2100:
  SCNCLR:GOSUB1500
495 CHAR18,3,"PAINTED CIRCLE":
  GOSUB2500
500 CIRCLE2,512,512,70,100:GOSUB
  2100:REGION7
505 REM PAINT SHAPES
510 PAINT2,512,512:GOSUB2100:
  SCNCLR:GOSUB1500
535 REGION0:CHAR18,3,"PAINTED
  SQUARE":GOSUB2500
540 DRAW2,400,450TO600,450:
  GOSUB2100
550 DRAW2TO600,650:GOSUB2100
560 DRAW2TO400,650:GOSUB2100
570 DRAW2TO400,450:GOSUB2100
580 REGION2:PAINT2,512,512:
  GOSUB2100:SCNCLR:GOSUB
  1500
605 REGION0:CHAR18,3,"PAINTED
  ELLIPSE":GOSUB2500
610 CIRCLE2,512,512,200,100:GOSUB
  2100
620 REGION6:PAINT2,512,512:
  GOSUB2100:SCNCLR:GOSUB
  1500
645 REGION0:CHAR18,3,"PAINTED
  TRIANGLE":GOSUB2500
650 DRAW2,550,450TO350,650:
  GOSUB2100
660 DRAW2TO750,650:GOSUB2100
  
```

```

670 DRAW2TO550,450:GOSUB2100
680 REGION5:PAINT2,512,512:
  GOSUB2100:REGION0
681 REM MUSIC MAKING
686 GRAPHIC0:GOSUB1200
  
```

```

687 PRINT"  SHIFT  and  CLR HOME
  CSRS CSRS CSRS CSRS
8 spaces MUSIC":GOSUB1300:
  GRAPHIC2:GOSUB3000
690 REM PLAY CHORD
691 SOUND225,225,225,0,9:FORG=1
  TO2000:NEXT:SOUND0,0,0,0,0:
  GOSUB3000
695 R=400
697 REM DRAW STAFF
700 DRAW2,1,RTO999,R:GOSUB2100
710 R=R+100:IFR>800THEN730
720 GOTO700
725 REM DRAW CLEF
730 CIRCLE2,200,900,7,14
740 DRAW2,200,900,TO200,325
750 CIRCLE2,200,400,52,75,75,25
760 CIRCLE2,200,663,130,196,25,75
770 CIRCLE2,200,700,100,147,75,25
775 CIRCLE2,200,625,60,80,5,75
780 CIRCLE2,250,660,7,14
790 GOSUB2500
795 REM DRAW NOTES
800 R=300:R1=900:A=0
805 REGION0
810 CIRCLE2,R,R1,28,40
812 DRAW2,R+28,R1OR+28,R1-200
815 A=A+1:ONAGOSUB915,920,
  925,930,935,940,945,950
820 R=R+90:R1=R1-50
830 IFR>1000THENGOSUB1000
840 GOTO810
855 REM PLAY NOTES

915 PRINT"  CTRL  and  ←
  V9T6S2O2C":RETURN

920 PRINT"  CTRL  and  ←
  V9T6S2O2D":RETURN

925 PRINT"  CTRL  and  ←
  V9T6S2O2E":RETURN

930 PRINT"  CTRL  and  ←
  V9T6S2O2F":RETURN

935 PRINT"  CTRL  and  ←
  V9T6S2O2G":RETURN
  
```

```

940 PRINT"  CTRL  and  ←
  V9T6S2O2A":RETURN

945 PRINT"  CTRL  and  ←
  V9T6S2O2B":RETURN

950 PRINT"  CTRL  and  ←
  V9T6S2O37C":RETURN
1000 FORG=1TO500:NEXT
1002 REM DISPLAY NOTES
1005 AAS="V9T4S2O2"
1006 REGION6
1007 CHAR4,6,"C":PRINTAAS"CR";
1010 CHAR4,7,"D":PRINT"DR";
1015 CHAR4,8,"E":PRINT"ER";
1020 CHAR4,9,"F":PRINT"FR";
1025 CHAR4,10,"G":PRINT"GR";
1030 CHAR4,11,"A":PRINT"AR";
1035 CHAR4,12,"B":PRINT"BR";
1040 CHAR4,13,"C":PRINT"T8O3C"
1050 FORG=1TO2000:NEXT:SCNCLR
1067 GRAPHIC0:END
1195 REM REMOVE SCREEN
1200 POKE36865,150:RETURN
1295 REM SCROLL SCREEN
1300 X1=36865:FORJ=150TO35STEP-1:
  POKEX1,J:FORG=1TO10:NEXTG,
  J:RETURN
1500 CHAR2,6,"ARTWORK":GOSUB
  2500:RETURN

2000 PRINT"  CTRL  and  ←
  V9T2S3O3C":FORJ=1TO500:
  NEXT:RETURN
2100 FORQQ=1TO500:NEXT:RETURN
2500 FORW=1TO1000:NEXT:RETURN
3000 CHAR2,7,"MUSIC":RETURN
30010 REM SUPER
30015 REM EXPANDER
30020 REM DEMO
  
```

# NOTES

# INDEX

Adding carriage returns	5	Grid	6
Alto voice	11,13	High-resolution graphics mode	7,8,9
Arc, drawing an	9	Indirect mode	2
Auxiliary color	2,6	Joystick, values returned by the	1,16,17
Axis, x and y	6	KEY	2,4
BASIC, compatability with VIC	1,3	Key, RESTORE	3,12
Bass voice	11,13	Keys, function	1,2
Border color	2,6	Light Pen, values returned by the	1
Carriage returns, adding	5	Lines, drawing	8
Cartridge, loading the SUPER		Loading the SUPER	
EXPANDER	1	EXPANDER cartridge	1
Changing function key assignment	4	Matrix screen	6,7
CHAR	2,9	Memory Expansion board, VIC1010	1
Character color	2,7,8,9	Memory, extra RAM	1,2
Character color, changing the	7,8	Mode, entering music	12
Character set, musical	12	high-resolution graphics	7,8,9
Chord, playing a musical	11	multi-color graphics	6,7,8,9
CIRCLE	2,3,9	Multi-color graphics mode	6,7,8,9
Circle, drawing a	9	Music character display, initializing	12
Clearing a graphics screen	10	terminating	12
COLOR	2,7,8,15	Music character set	12
Color, changing the character	7,8	Music mode, entering	12
Color registers	6,7,8,9,15	Octave selection	13
Color values	6	Origin, screen	6
COLOR	2	Paddles, values returned by the	1,16
Commands, entering SUPER		PAINT	2,7,9
EXPANDER	2	Painting graphics shapes	1,9
Commands, list of	2	Plotting graphics	1,6
Conventions, format	2	POINT	2,8
Coordinates, screen	6,8,9	PRINT	11,12,15
Debugging	17	Printing music	11,12
Direct mode	2	Programs, running SUPER	
Displaying music characters	12	EXPANDER	3
DRAW	9	Programs, terminating SUPER	
Drawing an arc	9	EXPANDER	3
a circle	9	RAM, extra memory	1,2
an ellipse	9	Range of screen coordinates	6
a line	8	RCOLR	15
Ellipse	9	Read functions	2,15,17
Entering single-note music mode	12	Rectangles, drawing	8
Errors, indication of	3	RDOT	2,16
Expansion, memory	2	REGION	2,7,15
Flat, playing a music note	14	Registers, color	6,7,8,9,15
Function keys:		sound	11,15
adding carriage returns to the	5	RESTORE key	3,12
changing the values of the	1,4,5	[RETURN]	2
preassigned values of the	2,4	RGR	2,15
Gradient	9	RJOY	16,17
GRAPHIC	2,7,15	RPEN	2,16,17
Graphics mode, high-resolution	7,8,9	RPOT	2,16,17
mixed	6,7,9	RSND	2,17
multi-color	6,7,8,9		

Screen, clearing a graphics	10
border	6,7
color	6,7,8
coordinates	6,8,9
matrix	6,7
origin	6
SCNCLR	2,10
Sharp, playing a music note	14
SOUND	2,11,15
Sound registers	11,15
Tempo, setting music	14
Termination of SUPER	
EXPANDER programs	3
Termination of music character	
display	12
Text display	6
Treble voice	11,13
Triangles	8
VIC 1010 Memory Expansion	
Board	1
Voice	11,13
Volume, setting multi-note	11
setting single-note	13
White noise, creating	11,13
Zero point of screen coordinates	6

## SUMMARY

(continued from back cover)

The parameters in the command formats are listed below:

n	an integer number
"entry"	a user-defined command
sc	screen background color
bo	screen border color
ch	character color
au	auxiliary color
cr	color register
x,y	coordinates of a point on the screen
c	color
rx	half the width of a circular shape
ry	half the height of a circular shape
as	starting point of an arc in gradians
af	finishing point of an arc in gradians
"text"	character string
s1-s4	sound voices
s5	volume level
p	a single music note in the range C to B
z	any number between 0 and 255
w	a number in the range 0 to 3
u	a number in the range 1 to 5
K	either 0 or 1

The colors and their associated values are listed below:

0	Black
1	White
2	Red
3	Cyan
4	Purple
5	Green
6	Blue
7	Yellow
8	Orange
9	Light Orange
10	Pink
11	Light Cyan
12	Light Purple
13	Light Green
14	Light Blue
15	Light Yellow

The commands that are assigned to the function keys when the SUPER EXPANDER cartridge is initialized are listed below:

KEY 1,	"GRAPHIC"
KEY 2,	"COLOR"
KEY 3,	"DRAW"
KEY 4,	"SOUND"
KEY 5,	"CIRCLE"
KEY 6,	"POINT"
KEY 7,	"PAINT"
KEY 8,	"LIST" + CHR\$(13)

# SUMMARY OF SUPER EXPANDER COMMANDS

*Abkürzung: Probleme mit Variablen, 1. Zeile: Graphic 2; Graphic 0*

Operation	Syntax	Page
Change function key assignments	KEY or KEY n, "entry" .....	4
Select graphics set	GRAPHIC n .....	7
Choose colors	COLOR sc,bo,ch,au .....	7
Plot point(s)	POINT cr,x,y or POINT cr,x,y,x1,y1 .....	8
Change character color	REGION c .....	8
Draw line(s)	DRAW cr,x,y TO x1,y1 or DRAW cr,x,y TO x1,y1 TO or DRAW cr TO x,y .....	9
Draw circular shape	CIRCLE cr,x,y,rx,ry .....	9
Draw arc	CIRCLE cr,x,y,rx,ry,as,af .....	9
Paint enclosed area	PAINT cr,x,y .....	9
Display text on a graphics screen	CHAR row,column,"text" .....	10
Clear a graphics screen	SCNCLR .....	10
Play a chord	SOUND s1,s2,s3,s4,s5 .....	11
Enter single note music mode	CTRL/left-arrow .....	12
Display music characters	P .....	12
Turn off music character display	Q .....	12
Set volume	V n .....	13
Select voice	S n .....	13
Choose octave	O n .....	13
Define duration of note(s)	T n .....	13
Play rest(s)	R .....	14
Play note(s)	p or ppppp .....	14
Play note following sharp	# .....	14
Play note following flat	\$ .....	14
Read graphic mode	RGR (z) .....	15
Read value in a color register	RCOLR (w) .....	15
Read color of a point on the screen	RDOT (x,y) .....	16
Read value of a sound register	RSND (u) .....	16
Read value indicated by a games paddle	RPOT (p) .....	16
Read value indicated by a light pen	RPEN (K) .....	16
Read value indicated by the joystick	RJOY (z) .....	17

The command KEY can only be entered in direct mode. The commands REGION and DRAW are only available in indirect mode. All other commands are available in both direct and indirect modes. Commands used to create graphics and the SOUND command are executed in direct mode by pressing the RETURN key. The remaining music commands are executed as the key associated with the command is pressed, after entering single note music mode. The read values must be preceded by the BASIC command PRINT in direct mode. In indirect (program) mode, single note music must be PRINTed.

Additional summary information on reverse of this page.

*Pixelmatrix 160 x 160*  
*Koordinaten: X : 0 000 1023 ca 6.39 Koordinaten*  
*Y : 0 000 1023 Punkt*

 **commodore**  
COMPUTER