£1.99

PHOENIX COMPUTER CRIB CARD

MEMOTECH MTX

KEYWORDS

OPERATING COMMANDS

GRAPHIC AND SOUND COMMANDS

COLOUR COMMANDS

DATA COMMANDS

INPUT/OUTPUT COMMANDS

BASIC STATEMENTS

BASIC FUNCTIONS

LOGICAL OPERATORS

ERROR MESSAGES

SPRITE COMMANDS

ASSEMBLER COMMANDS

EVERYTHING YOU NEED AT YOUR FINGERTIPS

KEY TO CONTENTS

addr	=	address
c	=	condition
cn	=	colour number
e	=	expression
i	=	integer
inc	=	increment
ln	=	line number
n	=	number
v	=	numeric variable
v$	=	string variable or literal string
x,y	=	coordinates
[ ]	=	optional

PROGRAM INSTRUCTIONS

KEYWORD	SYNTAX	DEFINITION
AUTO	AUTOln,inc	Auto line numbering:ln is first line number. inc is steps by which the numbers increase
CONT	CONT	Used directly to restart program after break key has been pressed
EDIT	EDITln	This causes the line to be copied to the edit screen
LIST	LISTln,ln1	Lists lines from ln to ln1
	LISTln	Lists lines from ln to end of program
LLIST	LLIST	Lists the program to the printer
NEW	NEW	Clears memory of program and variables
REM	REM	Remark or reminder
RUN	RUN	Executes a program

GENERAL INSTRUCTIONS

KEYWORD	SYNTAX	DEFINITION
BAUD	BAUDn,n1	Sets the RS232 channel n to baud rate n1
CLEAR	CLEAR	Clears variables
PAUSE	PAUSEn	Pauses the program for n/100 seconds
STOP	STOP	Stops the program

STRING FUNCTIONS

KEYWORD	SYNTAX	DEFINITION
ASC	ASC(v$)	Returns ascii code of v$
CHR$	CHR$(n)	Returns character with ascii code n
LEFT$	LEFT$(v$,n)	Returns n left most characters
LEN	LEN (v$)	Returns length of v$. N.B. the space is needed
MID$	MID$(v$,n,n1)	Returns a substring of n1 characters starting at the nth character
RIGHT$	RIGHT$(v$,n)	Returns n right-most characters
STR$	STR$(v)	Converts numeric expression to a string

NUMBER FUNCTIONS

KEYWORD	SYNTAX	DEFINITION
ABS	ABS(n)	Returns absolute value of n
EXP	EXP(n)	Returns e∧n
INT	INT(n)	Returns n truncated to an integer
LN	LN(n)	Returns natural logarithm of n
MOD	MOD(n,n1)	Returns the remainder of the division n/n1
RAND	RANDn	Seeds the random number generator
RND	RND	Returns a random number between 0 and 1 to nine decimal places
SGN	SGN(n)	Returns the signum of n
SQR	SQR(n)	Returns the square root of n

TRIGONOMETRIC FUNCTIONS

KEYWORD	SYNTAX	DEFINITION
ATN	ATN(n)	Returns arc-tangent of n
COS	COS(n)	Returns cosine of n
PI	PI	Returns closest representation of PI
SIN	SIN(n)	Returns sine of n
TAN	TAN(n)	Returns tangent of n

ARITHMETIC OPERATORS

Symbol	Operation	Priority
+	Addition	4
-	Subtraction	4
*	Multiplication	3
/	Division	3
( )	Brackets	2
∧	Exponentation	1

RELATIONAL OPERATORS

Symbol	Operation
=	Equal to
<	Less than
<=	Less than or equal to
<>	Not equal to
>	Greater than
>=	Greater than or equal to

LOOP,DECISION AND CONTROL STRUCTURES

KEYWORD	SYNTAX	DEFINITION
ELSE	IFcTHENaction ELSEaction1	If conditon is not met then action1 will be executed
FOR	FORv=nTOn1	Initiates FOR/NEXT loop.v =loop count,n=start value and n1=end value
GOSUB	GOSUBln	Calls subroutine at line ln
IF	IFcTHENaction [ELSEaction1]	Decision structure.If the condition is met, the action following the THEN statement is executed,if c is false, control passes to the next line,or the ELSE statement if included
NEXT	NEXT[v]	Terminates the loop when v=end value of the loop counter
ON	ONnGOSUB/ GOTO ln,ln1,...	Transfers control to the line corresponding to n.If n=1,then ln executed.If n=2,then ln1 is executed.etc
RETURN	RETURN	Returns control to program after subroutine
STEP	FORv=nTOn1 [STEP inc]	Specifies the value by which the loop counter is incremented
THEN	IFcTHENaction [ELSEaction1]	If condition is met,the THEN statement is executed
TO	FORv=nTOn1	Separates start and end values of loop counter

DATA HANDLING COMMANDS

KEYWORD	SYNTAX	COMMAND
DATA	DATAv,v$,..	Stores string or numeric data, separated by commas
DIM	DIMv(n,n,..) DIMv$(n,n,..)	Dimensions a string or numeric array, reserving storage space in memory
INKEY$	LETv$=INKEY$	Puts the character pressed on the keyboard into v$
INPUT	INPUT["prompt"] ,v$,..	Halts the program until the correct type of data has been entered (i.e.string or numeric)

LET	LETv=n
	LETv$=v$
		Defines numeric or string variable

READ	READv,v$ READv(n),v$(n)	Reads the data held in DATA statements into its string or numeric array/variable
RESTORE	RESTOREln	Restores the data pointer back to the beginning of line ln

FILE HANDLING COMMANDS

KEYWORD	SYNTAX	COMMAND
LOAD	LOADv$	Loads the named program v$ into memory from tape
	LOAD""	Loads the first program encountered on tape
SAVE	SAVEv$	Saves the program in memory onto tape
VERIFY	VERIFYv$	Verifies that the program on tape matches the one in memory

NODDY COMMANDS

KEYWORD	SYNTAX	COMMAND
ADVANCE	*A[DVANCE]	Removes the next program off the stack and executes it
BRANCH	*B[RANCH] v$	Branches to the section of program preceded by v$
DIR	DIR	Displays a directory of the Noddy programs and pages stored in memory
DISPLAY	*D[ISPLAY] v$	Displays the screen with the name v$
ENTER	*E[NTER]	Halts the program until an input is given
GOTO	*G[OTO] v$	Go to page v$ or label v$
IF	*I[F] v$,v1$	If input matches v$ the go to label v1$
LIST	*L[IST] v$	Print Noddy page v$ on printer
NODDY	NODDY	Takes you into Noddy
OFFSTACK	*O[FF[STACK]]	Removes the next program from the stack without executing it
PAUSE	*P[AUSE]	Pauses the program for 1 second
PLOD	PLOD v$	Runs a Noddy program from within BASIC
RETURN	*R[ETURN]	Return Noddy back to BASIC
STACK	*S[TACK] v$,v1$	Stacks the pages v$,v1$ etc.

GRAPHICS AND PRINTING COMMANDS

KEYWORD	SYNTAX	COMMAND
ANGLE	ANGLEn	Sets the initial orientation of the graphics in radians
ARC	ARCn,n1	Draws an arc length n while turning through an angle of n1 radians
ATTR	ATTRn,n1	Determines the effect on the graphics screen of using the plotting and printing commands. n=0 - characters are printed in paper colour on ink colour n=1 - characters are merged with those already present n=2 - plots the paper colour (unplot) n=3 - plots ink colour if paper was there and ink colour if paper was there before n1=0 - switches attribute off n1=1 - switches attribute on
CIRCLE	CIRCLEx,y,n	Draws a circle of radius n at the coordinates x and y
CLS	CLS	Clears the screen of graphics and text
COLOUR	COLOURn,cn	Sets the colours for the graphics screen n=0 - print paper is set n=1 - print ink is set n=2 - plot paper is set n=3 - plot ink is set n=4 - border colour is set
CSR	CSRx,y	Moves the cursor to the specified position
DRAW	DRAWn	Draws a line of length n from the current plot position in the direction set (see ANGLE and PHI)

GRAPHICS AND PRINTING COMMANDS

KEYWORD	SYNTAX	COMMAND

GENPAT	GENPATn,n1, n2,n3,n4,n5, n6,n7,n8,n9, n10	This command is used to generate all the patterns used for characters and sprites.
		n=0 - Redifines an ascii character. n1 32-127
		n=1 - Define a non ascii character. n1 129-154
		n=2 - Define colour for each line. n1 147-154
		n=3 - 8*8 sprite pattern n1 pat.no.
		n=4 - 16*16 NW quarter n1 pat.no.
		n=5 - 16*16 SW quarter n1 pat.no.
		n=6 - 16*16 NE quarter n1 pat.no.
		n=7 - 16*16 SE quarter n1 pat.no.
		n2 to n10 are values giving the contents of each row of dots of the character or sprite being defined

INK	INKcn	Selects the ink colour
LINE	LINEx,y,x1,x1	Draws a line from x,y to x1,y1
LPRINT	LPRINT	This has the same format as print, but sends the output to the printer
PAPER	PAPERcn	Selects the paper colour
PHI	PHIn	Adjusts the angle of orientation by n radians
PLOT	PLOTx,y	Plots a pixel (point) at the coordinates x,y
PRINT	PRINT[["item"] [;v][,v$]]	Prints at the current cursor position

COLOUR TABLE

0	-	Transparent
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

PUBLISHED FOR GENPAT USER CLUB
by
PHOENIX PUBLISHING ASSOC. LTD
14 VERNON ROAD BUSHEY HERTS WD2 2JL

ISBN 0 9465 7645 9

The author acknowledges the permission of the manufacturers to include copyright material from their manuals.

PHOENIX
PUBLISHING
ASSOCIATES

SPRITE COMMANDS

KEYWORD	SYNTAX	COMMAND

ADJSPR	ADJSPRn,
	n1,n2
		This command adjusts a value which has previously been assigned.n1 is the sprite number.n2 is the new value.
		n=0 - pattern n2 0-31(size 1)
		0-127(size 0)
		n=1 - colour n2 0-15
		n=2 - x pos n2 0-255
		n=3 - y pos n2 0-255
		n=4 - x speed n2 0-255
		n=5 - y speed n2 0-255

CTLSPR	CTLSPRn,n1	This command gives prior notice to the computer on the sprites you are to use
		n=0 - speed n1 1-255
		n=1 - distance n1 no.of pixels
		n=2 - no.of sprites n1 0-32
		n=3 - no.of circling sprites n1 0-32
		n=4 - plot sprite n1 0-32
		n=5 - no.of moving sprites. n1 0-32
		n=6 - magnitude and size n1=0 -8*8 mag 1
		n1=1 -8*8 mag 2
		n1=2 -16*16 mag 1
		n1=3 -16*16 mag 2

GENPAT	GENPATn,n1, n2,n3,n4,n5, n6,n7,n8, n9,n10	See GRAPHICS AND PRINTING COMMANDS

MVSPR	MVSPRn,
	n1,n2
		A general purpose command which combines 4 distinct functions.
		n1 is the sprite number.
		n2 should be in the range of the activities
		n=1 - Movement
		n=2 - Pattern selection
		n=4 - Redirect
		n=8 - Plot at centre
		Add together the n values if a combination of activities are required

SPRITE	SPRITEn, n1,x,y,n2, n3,cn	This command sets the original functions of the sprites n=sprite number (1-32) n1=pattern number (0-127 size 0,0-31 size 1) x,y=coordinates of start position n2=speed in x direction (-128 to 127) n3=speed in y direction (-128 to 127)

VIRTUAL SCREENS

KEYWORD	SYNTAX	COMMAND
CRVS	CRVSn,n1, x,y,n2, n3,n4	This command allows you to create your own virtual screen. n-Virtual screen reference number (0-7) n1-Type of screen (0-text,1-graphics) x,y-Co-ordinates of top left corner n2-Width of virtual screen in characters n3-Height of screen in lines n4-Width of screen (40-text,32-graphics)
DSI	DSI	This command allows you to roam about freely within a screen only ending when the carriage return is pressed
EDITOR	EDITORv$	The editor allows the programmer to accept input from a defined area of the screen.The area of the screen is set by virtual screen 0 using the CRVS command
GR$	GR$x,y,n	This command reads a bit pattern from a graphics screen returning the value as a character.x and y are locations on the virtual screen.n is the number of bits to be read.
SPK$	LET v$=SPK	This command peeks the screen and gives the character at the cursor location on the current text screen
VIEW	VIEWn,n1	This command moves the window relative to the sprite planes whilst leaving the position of the sprites unchanged
VS	VSn	This command selects a virtual screen from those already defined or created using the CRVS command

The locations from FF5D hex to FFD4 hex contain data controlling virtual screens.These values may be changed using CRVS,POKE or from assembler code. Each virtual screen has a block of 15 bytes related to it.The format for these bytes is shown below:-

Byte	Contents
1	Various bits indicating screen type,auto scroll, cursor flash,page mode
2+3	Current PRINT position within this screen in column,row format.
4+5	Absolute top left hand corner,represented as above
6+7	Width of screen and no.of lines in characters
8	Line width of physical screen
9	Contains the character under the cursor
10	Shows border colour
11	Shows PRINT colours as ink,paper
12	Shows PRINT attributes
13	Shows PLOT colours as INK,PAPER
14	Shows PLOT attributes
15	Counter for number of lines scrolled so far

MEMORY COMMANDS

KEYWORD	SYNTAX	COMMAND
CLOCK	CLOCKv$	The clock is initiated to the value of the string
OUT	OUTn,n1	Outputs the specified value to the specified port
PEEK	LETv=PEEKaddr	Gives the contents of the specified address in the current page
POKE	POKEaddr,n1	The POKE command loads the specified memory location with the specified value
ROM	ROM 2	Passes control to an additional ROM pack, for example PASCAL or FORTH
TIME$	TIME$	This command is used to print or assign the time on the real time clock
USR	USR(addr)	USR causes control of the program to be transferred to the specified memory address

SOUND COMMANDS

KEYWORD	SYNTAX	
SBUF	SBUFn	This command makes space in a sound buffer for use by the SOUND command. The command makes space for n blocks of sound data for each of the channels.
SOUND	SOUNDn,n1,n2,n3, n4,n5,n6	The effect of the sound channel depends on the number of expressions used n-channel number (0-3) n1-frequency (0-1023) n2-volume (0-15) n3-frequency gradient (-32767 to +32767) n4-volume gradient (-32767 to +32767) n5-time (0-65535) n6-mode (0-1)

CONTROL CODES

RST 10 and Control Codes

ASCII	FUNCTION
1	PLOT X,Y
2	LINE X1,Y1,X2,Y2
3	CURSOR X,Y
7	BELL
10	LINE FEED,CURSOR DOWN
11	VERTICAL TAB
12	CLS/HOME
13	CARRIAGE RETURN
14	CTLSPR P,X
15	GENPAT P,N,D0,D1,D2,D3,D4,D5,D6,D7
16	COLOUR P,N
17	ADJSPR P,N,V
18	SPRITE N,P,XP,YP,XS,YS,COL
19	MOVSPR P,N,D
20	VIEW DIR,DIS
21	INSERT KEY
22	DELETE KEY
23	BACK TAB

CONTROL CODES

RST 10 and Control Codes

25	TAB KEY
26	HOME KEY
27,65	ATTR P,STATE
27,89	CRVS N,T,X,Y,W,H,S
27,90	VS N
27,67	GR$ X,Y,B (RESULT IN WORK SPACE)

CTRL A	PLOT X,Y
CTRL B	LINE X1,Y1,X2,Y2
CTRL C	CURSOR X,Y
CTRL Dn	Sets background colour to n
CTRL E	Erase to end of line
CTRL Fn	Sets foreground colour to n
CTRL G	Sounds the bell
CTRL H	Backspace,cursor left
CTRL I	Tabulate the next block of eight columns
CTRL J	Line feed,cursor down
CTRL K	Cursor up
CTRL L	Clear screen and home cursor
CTRL M	Carriage return,cursor to left edge of screen
CTRL N	CTLSPR P,X
CTRL O	GENPAT P,N,D1,D2,D3,D4,D5,D6,D7,D8
CTRL P	COLOUR P,N
CTRL Q	ADJSPR P,N,V
CTRL R	SPRITE N,P,XP,YP,XS,YS,COL
CTRL S	MOVSPR P,N,D
CTRL T	VIEW DIR,DIS
CTRL U	INSERT KEY
CTRL V	DELETE KEY
CTRL W	BACK TAB
CTRL Y	TAB KEY
CTRL Z	HOME KEY
CTRL]	Page mode
CTRL \	Scroll mode
CTRL ^	Cursor on
CTRL _	Cursor off

ASSEMBLER COMMANDS

ASSEM	ASSEMIn	Switches to the assembler and assembles the code at the specified line.
T	T	Takes the program pointer to the top of the program

Insert mode

<RET>	Enters at the program pointer position
£n	Enters at the hexidecimal address n
n	Enters at the decimal address n
label	Enters at the label if it exists

Edit mode

E	Enters the editor at the program pointer
E £n	Enters the editor at the hexidecimal address n
E n	Enters the editor at the decimal address n
E label	Enters the editor at the label position if it exists

ASSEMBLER COMMANDS

List mode

L	Lists the program from the program pointer
L £n	Lists the program from the hexidecimal address n
L n	Lists the program from the decimal position n
L label	Lists the program from the label if it exists

Print mode

P	Prints the program from the program pointer
P £n	Prints the program from the hexidecimal address n
P n	Prints the program from the decimal address n
P label	Prints the program from the label if it exists

FRONT PANEL COMMANDS

PANEL PANEL Switches on the front panel

Input	Result
B	Exit? Press Y to return to BASIC
C	Clears the list screen
D	Display> Entry of a hex number displays a block around that address
G	Go> A hex number must be entered To> A hex number must be entered This runs a program between the two hex numbers
I	Toggles between an ascii and hexidecimal memory display
L	List> Press the fullstop to give a listing from the program pointer A hex number gives a listing from that memory address
M	Move> A hex number must be entered End> A hex number must be entered To> A hex number must be entered This moves a block of memory to the new position
R	Register> A hex number must be entered This changes the value of the register to the new hex number
S	This is used to single step through the program by executing the command at the program counter
T	As S but calls are treated as a single instruction
X	Display the alternate register set
.	Move the register cursor
-	Move the display cursor backward
<RET>	Move the display cursor forward
Cursor up	Move the display cursor up
Cursor down	Move the display cursor down

ERROR MESSAGES

Error message	Meaning
Params	Incorrect or wrong number of parameters for a function or a command
Mistake	A mistake has been made which should be obvious from the context
A	Point outside the virtual screen
SE.A	Screen type not in type table
SE.B	Invalid ESC sequence
SE.C	Command not valid for this device
SE.D	Switch to absent virtual screen
SE.E	Invalid UDG type
Symbol?	A symbol is missing such as "=","TO","THEN" or ","
Not numeric	A number is expected
Not a string	A string is expected
Boolean?	A truth value is expected
Mismatch	An illegal relationship between different types of values
BK	Breaks in tape SAVE or LOAD
No data	No data for READ or no page for NODDY
Overflow	Number too big
Div/0	Division by zero
Out of range	Number is not in a valid range
No space	No space left to define an array,expand a program, perform a large operation or assign a string to a character array
Subscript	A subscript is out of range or there are too many.
Gosub	More than the maximum of 34 GOSUBs
Undefined	A variable is being used before it exists
Array exists	An array has already been defined
No FOR	A next has been encountered without a matching FOR
No call	A RETURN has been encountered without a matching GOSUB
No line	A reference has been made to a non-existent line

ESCAPE CODES

ESC 'A'	ATTR P,STATE
ESC 'Y'	CRVS N,T,X,Y,W.H.S
ESC 'Z'	VS N
ESC 'C'	GR$ X,Y,B (RESULT IN WKAREA)
ESC B0	American character font
ESC B1	English character font
ESC B2	French character font
ESC B3	German character font
ESC B4	Swedish character font
ESC B5	Spanish character font
ESC I	Inserts a blank line at cursor line
ESC J	Deletes the current cursor line
ESC K	Duplicates a line
ESC Xc	Simulates control character 'C'