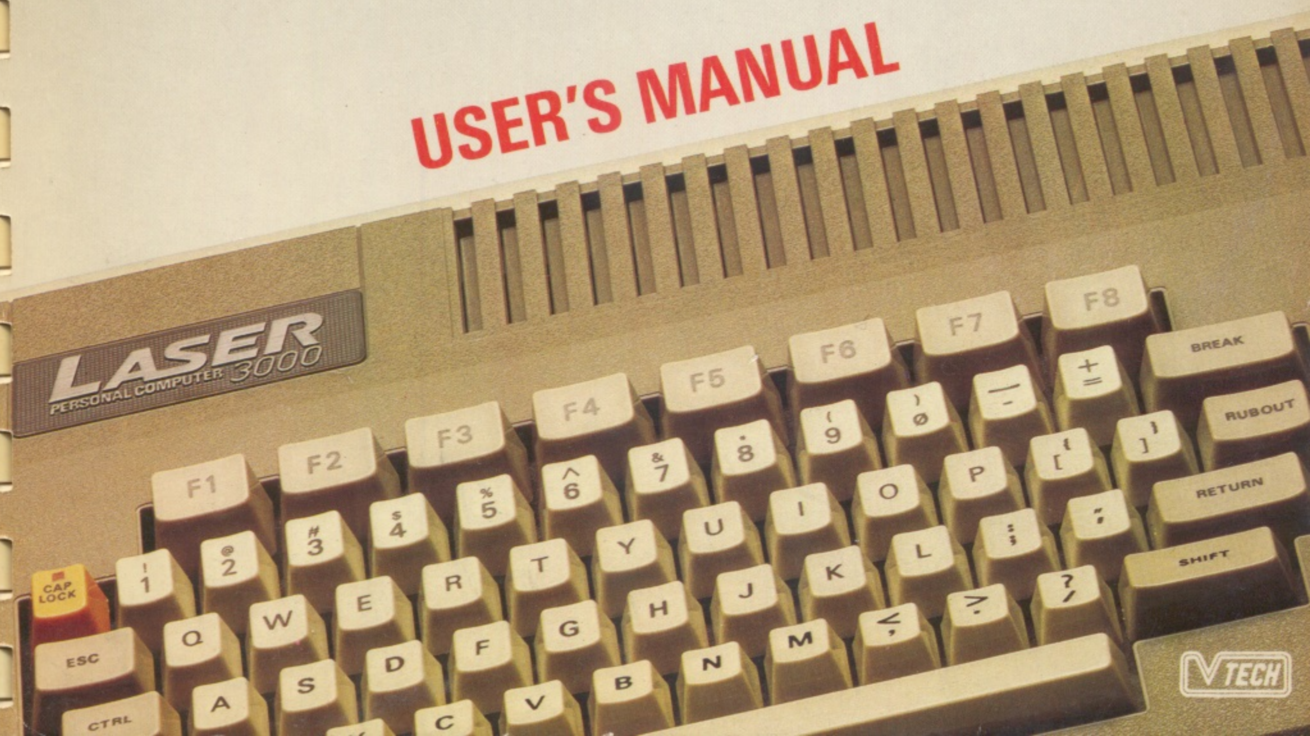
LASER™ 3000
PERSONAL COMPUTER

USER'S MANUAL



FIRST EDITION — 1984

CONTENTS

CHAPTER 1

GETTING STARTED

CHAPTER 2

A DESCRIPTION OF THE LASER 3000 KEYBOARD

CHAPTER 3

LASER 3000 DISPLAY MODES

CHAPTER 4

SCREEN EDITING FEATURES

CHAPTER 5

PLUG-INS AND ADD-ONS

CHAPTER 6

MEMORY ORGANISATION

CHAPTER 7

THE SYSTEM KERNEL

APPENDICES

CHAPTER 1

GETTING STARTED

GETTING STARTED

A FEW BRIEF WORDS...

Congratulations on buying the Laser 3000!

We consider that you have bought one of the finest home computers available today.

The Laser 3000 has graphics capabilities that are probably unsurpassed among the current home computers – and not surpassed by many other much more expensive personal computers either!

As well as its superb graphics – which you can see for yourself by playing the demonstration cassette tape which is packed with each unit – the Laser 3000 can run an enormous variety of software programs.

With the addition of expansion units, the Laser 3000's repertoire of software can be even further extended.

The use of your LASER 3000 is virtually unlimited; ranging from the serious bussiness applications to playing arcade-like games. The myriad of other uses will occur to you as your acquaintance with the Laser 3000 deepens into friendship.

Because the LASER 3000 is APPLE® II compatible, you now have access to the world's largest software library – over 4000 programs in all.

ADDENDUM

1. On end of page 9, add:

   (A) 50/60 Hz SWITCH, located on the bottom of the chasis (see fig A), allows you to match the synchronizing frequency of your display monitor or TV. Select the proper frequency to give you a stable display.

   (B) 40/80 COLUMN SWITCH, (Fig B) also on the bottom, allows you to select or de-select the built-in 80-column firmware.

   The normal position of this switch is 80 COL. However, you may have to choose 40 COL. in case 40 COL. Text or 280 X 192 graphics is needed. Select before power up.

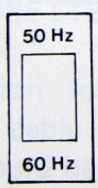


Fig. A

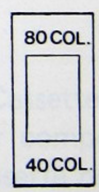


Fig. B

91-172-02

SETTING UP THE LASER 3000

IMPORTANT: PLEASE MAKE SURE THAT ALL COMPONENTS ARE TURNED OFF BEFORE YOU MAKE ANY CONNECTIONS.

To set up the system, you will need a display monitor, whether composite video type or RGB type. If you choose to use your home TV for display, you will need the TV20 TV interface unit. To run the demonstration tape, you will need an audio cassette player.

First, we will take a look at the back panel where you will make all the connection points. You will see. At the back panel, sockets, buttons and switches as in figure 1.1

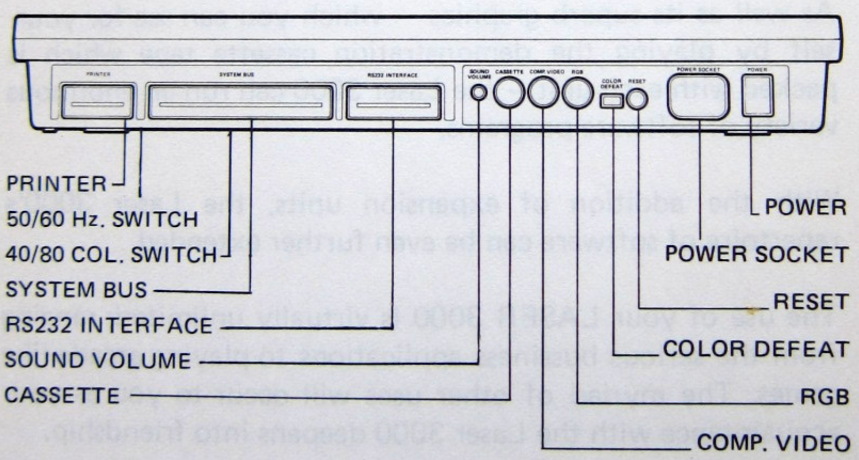


Figure 1.1 The Back Panel.

- the POWER SOCKET and the ON/OFF SWITCH

  this is the MAINS input to your LASER 3000. You should find the power cord in the packing.

- the COMPOSITE VIDEO SOCKET

  connection for a video monitor with composite video input, or for the TV interface unit if you are using a TV set for display. A composite video cable is supplied with your computer. The red plug should be connected to the monitor video input, while the black one plugged into the audio input, if the monitor has an internal speaker itself.
  note: if you are using a TV for display. You may not obtain satisfactory results from high-resolution or 80-column display, owing to the limited resolution on a domestic TV set.

- the CASSETTE SOCKET

  connection for the DRIO Data Cassette player. The cassette cable is supplied with your computer. The red plug should be connected to the cassette output and the black plug to the cassette input

- the RESET BUTTON

  pressing RESET will halt any program execution, and usually returns you to BASIC.

- the COLOUR DEFEAT switch

  when activated, will kill the *chroma* carrier and output only blcak and white signals to the composite monitor and TV displays.

- the SYSTEM BUS

  this contains all address, data and control signals, it is mainly used for system expansion.

- the PRINTER BUS

  connection for a printer with a CENTRONICS insterface.

- the RS232 bus

  connection for any pheripheral with a RS232 Serial Adaptor interface, auch as a modem or a serial printer.

- 50/60 Hz SWITCH this is located on the under side of the console, it allows you to match the synchronizing frequency of your display monitor or TV. Select the proper frequency to give you a stable display.

- 40/80 COLUMN SWITCH

  also on the under side of the console , it allows you to select or de-select the built-in 80-column firmware.

The normal position of this switch should be at 80 COL. However, you may have to choose 40 COL. in case the 80-column display memory is used for program storage, and 80-column display is not desirable. DO NOT flip the switch when power is on.

- SOUND VOLUMN CONTROL

this controls the sound output from the internal speaker. Turning it clockwise increases the volumn.

Now that you are ready. Make all necessary connections as shown is fig 1.2. Check before you switch on anything!

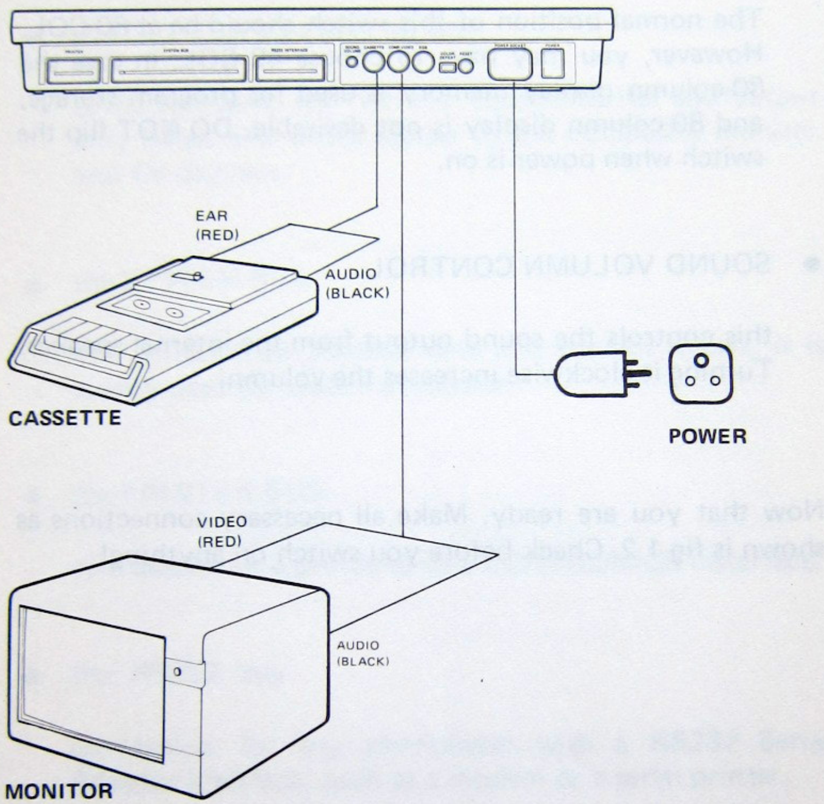


Figure 1.2 Making the wiring connections.

Now with the volume control set at the center postion (just in case you'll get a shock), switch on the display, monitor or TV (if you are use a TV, you'll need to tune it to the proper channel). Then switch on the LASER 3000.

UP UP AND AWAY!

After switching on, you will hear a beep and the display sholud look something like that in figure 1.3.

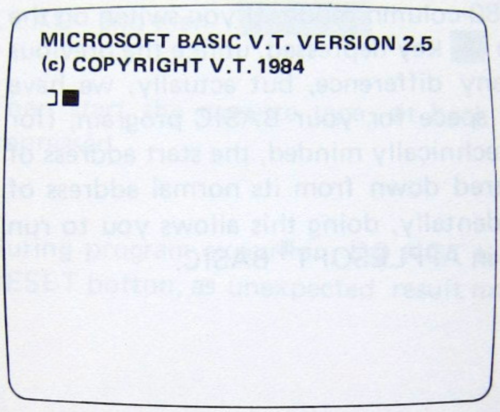


Figure 1.3 The screen at power up.

If you hear or see anything different, switch off all components and check the cable connections carefully. Consult your local dealer if you have any further problems.

POWER-UP TRICKS

Just to mention, there are some power-up 'tricks' you can perform on the LASER 3000. If you switch on the computer with the ESC key depressed, the display will be automatically switched to 80-column mode. If you switch on the LASER 3000 with the A key depressed, unlike the previous case, you won't see any difference, but actually, we have created more memory space for your BASIC program, (for those of you who are technically minded, the start address of basic programs is lowered down from its normal address of $1801 to $0801). Incidentally, doing this allows you to run more programs written in APPLESOFT® BASIC.

RUNNING THE DEMONSTRATION TAPE

To load the demonstration programs from the supplied cassette, rewind the tape to the start then type in:

LOAD and then RETURN

Then start the cassette tape, sit back and get ready to be impressed.

During program execution, DO NOT hit the BREAK or RESET botton, as unexpected result may occur.

CARING FOR YOUR LASER 3000

Your Laser 3000 is a delicate, precision equipment and just like your home television or high fidelity audio components, it should be treated with care.

- Never subject it to heavy mechanical shock such as dropping it from height.
- Make sure that no moisture gets into it.
- Avoid cigarette ash or excessive dust.
- Do not block the air ventilation holes.
- Place your computer away from direct sunlight or sources of intense heat or cold.

WARNING

Never open your computer as this will void your warranty and expose you to electrical hazard.

CHAPTER 2

A DESCRIPTION OF THE LASER 3000 KEYBOARD

A DESCRIPTION OF THE LASER 3000 KEYBOARD

Through the keyboard you can talk to the LASER 3000. Although it may not understand what you say once in a while, it will never damage the computer by typing garbage in (unless you are use to typing with a sledge-hammer).

The keyboard layout of the LASER 3000 is as shown in figure 2.1.

Figure 2.1 Laser 3000 Keyboard.

The keyboard is divided into two parts.

The lefthand side resembles a typewriter's, and the righthand one is known as a numeric keypad. This numeric keypad is mainly used when you are entering a lot of numbers.

THE MAIN KEYBOARD

Apart from the standard typewriter keys, there are a number of special keys in the keyboard. These can be distinguished by their colour difference.

RETURN

The RETURN key implies 'Return control to the computer'. After you hit this key, the computer will start interpreting whatever you have typed in before RETURN . If the computer does not understand you at all, it will mildly show its annoyance by giving you a 'beep' and an error message.

Hitting RETURN has another effect; it performs a 'carriage return' on the screen, i.e., it will put the little flashing cursor at the beginning of next line down, or it will cause the screen to scroll up if the display has already reached the bottom of the screen.

Figure 2.2 The RETURN Key.

SHIFT

On either side of the main keyboard there are two key labelled SHIFT . These perform the same function as that on an ordinary typewriter.

They do not generate any character code when used on its own. When pressed with another key, a different character code will be send to the computer. For instance, pressing SHIFT and 7 will enter a '&' into the computer.

Figure 2.3 The SHIFT Key.

CAP LOCK

From the keyboard, you can enter either upper or lower case letters to the LASER 3000. The orange CAP LOCK key is a electronic locking switch (CAPital Lock). It toggles between capital and lower case letter entry. When the LASER 3000 is first turned on you may notice that the red light on the CAP LOCK key is on, indicating that all letters typed will be in capital.

Press the key once will turn off the light and change to lower case entry. Pressing it once again will revert to capital entry.

Figure 2.4 The CAP LOCK Key.

TAB

The orange key right next to the space bar is the TAB key.

Hitting the TAB key will move the cursor to the nearest TAB column to the right of the cursor.

The TAB columns are 8, 16, 24, 32, 40, 48, 56, 64 and 72, in 80-column display. While in 40-column mode, they are 8, 16, 24, 32.

So, if the cursor is positioned in column 16 for instance, pressing TAB will move it to column 24.

Figure 2.5 The TAB Key.

F1 F2 F3 F4 F5 F6 F7 F8

Across the top of the main keyboard is a row of eight keys labelled F1, F2, F3, F4, F5, F6, F7 and F8.

These are function keys, they allow you to enter commands or sequence of commands with a single keystroke (yes, even computer freaks are lazy sometimes).

Each function key is actually three-in-one. Each one of them is accessed either by pressing the key alone, or together with the SHIFT key, or together with the CTRL key. So, all in all, there are 24 function keys!

Table 2.1 is a list of the 24 functions you get from the function keys, when you first switch on the LASER 3000.

F1 to F8 are obtained by pressing the keys alone.

F9 to F16 are obtained by pressing one of the function keys together with the SHIFT key.

F17 to F24 are obtained by pressing the CTRL key together with one of the eight keys.

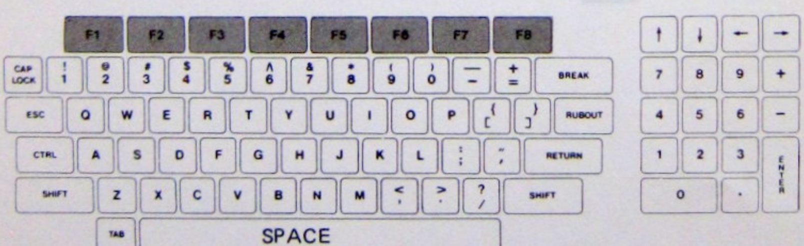


Figure 2.6 The FUNCTION Key.

Function No.	Function
F1	LIST RETURN
F2	RUN
F3	HOME RETURN
F4	TEXT RETURN
F5	TEXT NORMAL,, BLACK RETURN
F6	WIDTH 80 RETURN
F7	WIDTH 40 RETURN
F8	PR#8 RETURN
F9	CALL-151 RETURN
F10	SOUND DEF
F11	SOUND TEMPO
F12	SOUND
F13	DRAW SCIRCLE (
F14	DRAW HSCIRCLE (
F15	DRAW SSQUARE (
F16	DRAW HSQUARE (
F17	PR#1 RETURN
F18	HGR
F19	HCOLOR=
F20	HPLOT
F21	PAINT (
F22	PR#6 RETURN
F23	NOISE
F24	PRINT USING

Table 2.1 Function key default functions.

You can change the function key definitions using the IN#8 command (See BASIC REFERENCE MANUAL.).

To see what the definitions are just type PR#8 and RETURN. Those of you who are observant might noticed F8 will do just that.

As a additional bonus, you can save your function key definitions onto the disk by typing:

BSAVE<anyname you fancy>, A$800, L$200 and RETURN
Every time you want to recall those definitions, type:

BLOAD< the same name you used before>and RETURN

For a more detailed explanation of these commands, see the LASER 3000 Dos manual.

If the display moves too fast for you, enter CTRL and S to stop it, and then any key whatsoever to continue the scrolling.

ESC

Pressing the ESC key on the left of the keyboard will not make the LASER 3000 Escape and turns into a wild beast. What it really does is to give a totally different meaning to all keys you press thereafter (know as escape sequence). Table 2.2 shows the use of this ESC sequence. To escape from the escape mode (Sorry!), just press ESC again.

Figure 2.7 The ESC Key.

ESC	I	move cursor up
ESC	J	move cursor left
ESC	K	move cursor right
ESC	M	move cursor down
ESC	@	clear screen
ESC	E	clear from cursor to end-of-line, then leave ESC mode
ESC	F	clear from cursor to end-of-screen, then leave ESC mode
ESC	A	right cursor and leave ESC mode
ESC	B	left cursor and leave ESC mode
ESC	C	down cursor and leave ESC mode
ESC	D	up cursor and leave ESC mode
ESC	G	place the cursor at HOME position, i.e. top left hand corner

Table 2.2 Escape sequences.

CTRL

The control key, located above the ESC key, does not generate a key code of its own, but only alters the codes of other keys pressed. Some of the keys when pressed with the CTRL key, will perform special functions as per table 2.3.

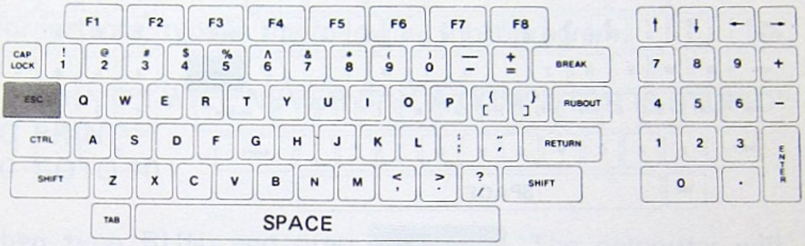


Figure 2.8 The CTRL Key.

CTRL	G	sound the beeper.
CTRL	H	backspace
CTRL	J	line feed
CTRL	K	clear the screen from current cursor position to end of the display screen.
CTRL	L	clear the whole screen and put cursor at the 'HOME' position.
CTRL	M	return
CTRL	S	stop display from scrolling upwards: press any other key to resume scrolling.
CTRL	U	right cursor
CTRL	Y	home the cursor.

Table 2.3 Control functions.

RUBOUT

Immediately above the RETURN key is the RUBOUT key. This key works as an electronic eraser and removes the character immediately to the left of the cursor.

Figure 2.9 The RUBOUT Key.

BREAK

The BREAK key is located just above the RUBOUT key. Pressing BREAK will stop the program at the end of the executing statement and will output on to the screen a message: BREAK IN nn, where nn is the number of the statement just completed.

For instance, type in the following short program.

```
10 PRINT"I WILL NEVER SAY A COMPUTER IS STUPID"
20 PRINT
30 GOTO 10
```

Then type RUN, and press RETURN. The computer will then print out the appropriate message continuously. Pressing BREAK now will halt program execution (typing CTRL and C will have the same result).

To continue the program you have just interrupted, type in CONT and press RETURN.

Figure 2.10 The BREAK Key.

THE NUMERIC KEYPAD

cursor movement keys, which allow you to move the cursor to anywhere on the screen.

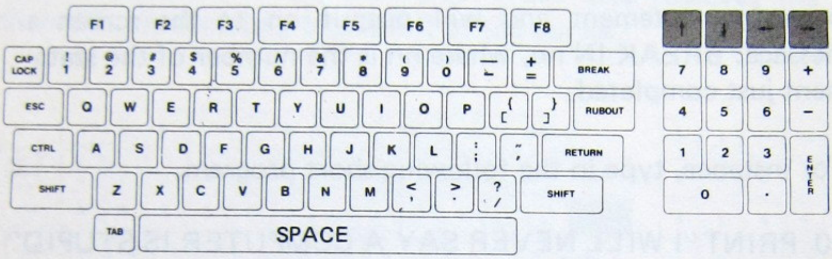


Figure 2.11 The cursor control keys.

The NUMERIC KEYS are for speeding up the entry of numbers. The ENTER key is identical in function to the RE-TURN key on the main keyboard.

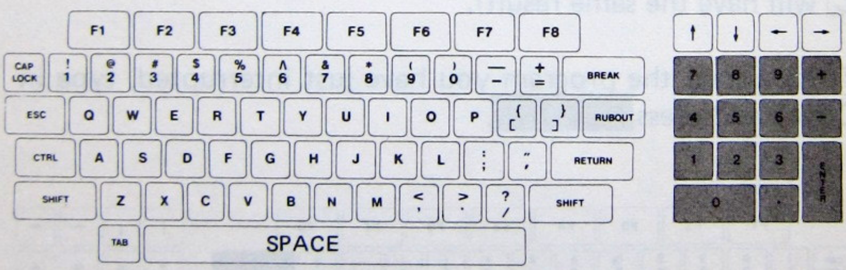


Figure 2.12 The numeric keys.

AUTO-REPEAT KEYS

All keys on the LASER 3000 keyboard have auto-repeat feature. If you press a key and hold it down for about a second, it will repeat itself at about 8 characters per second until you release it.

CHAPTER 3

LASER 3000 DISPLAY MODES

LASER 3000 DISPLAY MODES

There are five different display modes on the LASER 3000, they are listed below.

- 40-column text mode
- 80-column text mode
- low resolution graphics
- bit image graphics
- double resolution graphics

TEXT MODES

In text mode, the LASER 3000 can display 24 lines of characters with either 40 or 80 characters on each line. A total of 128 characters are included in the character set.

Each character is form on the display in a dot matrix (or pixel matrix, if you prefer to be technical) 7 dots wide and 8 dots high. Text characters are usually limited to only 5 dots wide and 7 dots high, to separate adjacent characters and adjacent lines.

Diagram 3.1 shows how the letter A and the number 8 are formed in the matrix.

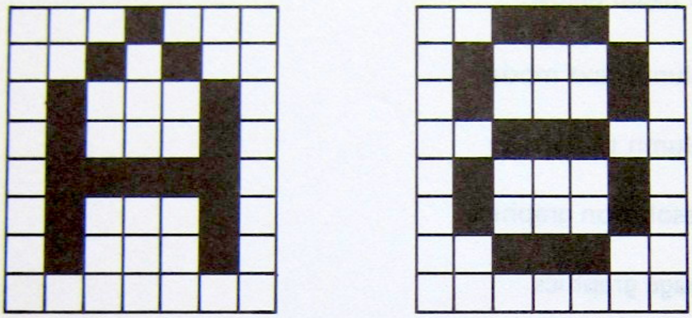


Diagram 3.1 The composition of a display character.

As a matter of interest, there are two displayable 'pages' in this mode:- the primary and secondary text page. The secondary page are usually hidden away, but can be displayed by selection of certain software switches. (see Chapter 6).

NORMAL TEXT

The Laser 3000 normally displays characters made up of white **pixels** on a dark background. However, you can choose from among eight colours for the characters, and then another for their background, and another still for the border framing the display screen.

To see this in action, enter:

TEXT RED, GREEN, BLUE

and then hit **RETURN**.

This sets the colour of the characters as red, the background as green, and the border as blue.

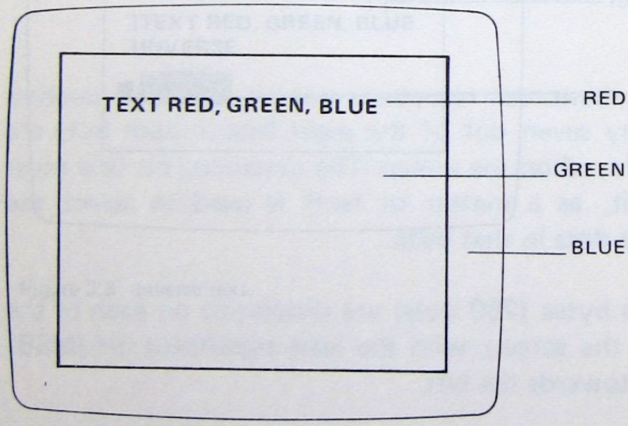


Figure 3.2 Normal text.

INVERSE TEXT

Inverse text is the switching of the background colour with the character colour. With your Laser display set up as in the previous example, now key in:

INVERSE

(not forgetting to hit RETURN), you should see all characters displayed thereon are green on a red background.

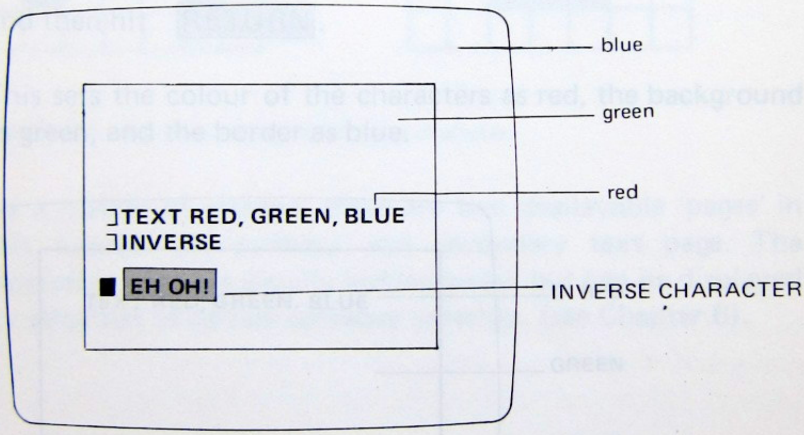


Figure 3.3 Inverse text.

FLASHING TEXT

You can also make the characters displayed to alternate between normal and inverse mode, i.e. flashing. Just enter:

FLASH and RETURN

Owing to hardware limitation, not all characters can flash, see the Basic Reference Manual and try it yourself.

LOW RESOLUTION GRAPHICS

In this mode your display screen is arranged in 192 horizontal lines of 280 **pixel**s each, which makes for a grand total of 53,760 pixels altogether. The graphic data is taken from a speical area of memory – the graphics memory.

Each dot on the screen represents one bit from the graphics memory. Only seven out of the eight bits in each byte are actually displayed on the screen. The remaining bit (the most significant bit, as a matter of fact) is used to select the colours of the dots in that byte.

A total of 40 bytes (280 dots) are dislplayed on each of the 192 lines of the screen, with the least significant bit (MSB) of each byte towards the left.

Some of you might have noticed the restraint of colour usage in this mode of display. Although there are six colours available in this graphics mode, their usage is subject to the following limitations:

1. All bits that are 'off' (equal to 0) have corresponding **pixels** in black.

   If a bit is 'on' (equal to 1), then.

2. If the **MSB** of a byte is 'off', then the **pixels** controlled by that byte are magenta if they occupy even columns, and green if they occupy odd columns.

3. If the **MSB** of a byte is 'on', then the **pixels** controlled by that byte are blue if they occupy even columns, and red if they occupy odd columns.

4. Two dots side by side will both appear white.

Confused? Well, type in the following program and see if we can make things clearer.

```
10    HGR

20    HCOLOR=3

30    INPUT X

40    HPLOT X,0 TO X, 191

50    GOTO 30
```

Now enter:

```
RUN
```

Type RUN and RETURN to execute the program. Just enter a number between 0 and 279 (then RETURN), notice the effect. You can change line 20 to:

```
20 HCOLOR=7
```

and execute the program again; you should get a different set of colours. (What do you mean by you don't have a colour monitor!).

In low resolution graphics mode, six colours are available, each has a different colour code, as shown in table 3.1.

number	colour
0	black
1	green
2	magenta
3	white
4	black
5	red
6	blue
7	white

Table 3.1 Colour Codes in double and low resolution graphics

Other than the two displayable pages with this mode-Primary and secondary page, there exists a special mixed text/graphic mode. In this mode the low resolution graphics window is reduced to allow 4 lines of textual display at the bottom of the screen. All these can be accessed via basic commands, HGR1, HGR2 and HGR respectively.

BIT IMAGE GRAPHICS

This graphics mode has the same resolution as low resolution graphics, i.e. 280 X 192 pixels. but there is no limitation on the colours that each pixel can assume.

In fact, each **pixel** can take on one of eight colours: white, red, green, blue, cyan, yellow, magenta, and black.

Also, a **pixel** can have any of these colours irrespective of its position.

Try the following program:

```
10  HGR 5
20  R1 = RND (1) * 180
30  R2 = RND (1) *92
40  X = 50 + R1
50  Y = 50 + R2
60  RADIUS = RND (1) * 49 + 1
70  HCOLOR = RND (1) * 6 + 1
80  DRAW SCIRCLE (X, Y), RADIUS
90  GOTO 20
```

If you have seen enough of the display, depress BREAK to stop the program, then type TEXT to return to text mode.

A total of eight colours are available in the bit image graphics mode. Their corresponding codes are shown it table 3.2.

code	colour
0	black
1	green
2	magenta
3	cyan
4	yellow
5	red
6	blue
7	white

Table 3.2 Bit image graphics colour code.

The primary and secondary page of the bit image graphic mode can be accessed via the HGR5 HGR6 basic commands respectively.

DOUBLE RESOLUTION GRAPHICS

As its name suggests, this graphics mode has twice the resolution of the previous two graphics modes.

Across each line there are 560 **pixels**, and there are 192 lines, giving a total of 107,520 **pixels** in all.

The colouring rules and colour codes for this mode follows that for low resolution graphics.

Type in the following program carefully, if you find no mistake, then type RUN and RETURN.

```
10  HGR3
20  A1 = RND (1) * 279
30  A2 = A1 + 279
40  B = RND (1) * 191
50  I% = (RND (1) * 4) + 2
60  FOR X = 0 TO 278 STEP I%
70  FOR S = 0 TO 1
80  HCOLOR = 3 *S
90  HPLOT X + S, 0 TO A1, B TO 279 -X - S, 191
100 HPLOT X + 279 + S, 0 TO A2, B TO 558 - X - S, 191
110 NEXT S, X
120 FOR Y = 0 TO 190 STEP I%
130 FOR S = 0 TO 1
140 HCOLOR = 3 * S
150 HPLOT 279, Y + S TO A1, B TO 0, 191 - Y - S
160 HPLOT 558, Y + S TO A2, B TO 279, 191 - Y - S
170 NEXT S, Y
180 END
```

CHAPTER 4

PLUG-INS AND ADD-ONS

SCREEN EDITING FEATURES

THE LEFT AND RIGHT ARROW KEYS

The ← key is actually a backspace key. It moves the cursor left one space. It erases the character it passes over from the line you are editing (notice that it does not wipe out the character on the screen.) If you haven't pressed RETURN at the end of the last line you typed, the ← key only deletes the characters on that line.

The → key, on the other hand, is a retype key. It moves the cursor right one space and retype the character it passes over. Moving the cursor along a line using the → key has the same effect as typing that line out again.

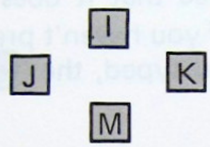
THE UP AND DOWN ARROW KEYS

Unlike the left and right keys, these two keys are pure cursor movement keys. They are used to move the cursor without affecting any characters on the screen.

Using these four keys, you can edit your program lines alike by moving the cursor to the appropriate place, then retype or correct the line as necessary.

PURE CURSOR MOVES

The ESC , I , J , K , M keys are used to move the cursor to anywhere without affecting any of the characters it happens to pass over. The keys represent, respectively, up, left, right and down cursor moves. You'll find the keys are arranged on the keyboard in a easy-to-remember star configuration.



Pressing ESC once will get you into editing mode, use the mentioned keys to move the cursor to the appropriate place. To exit the edit mode, press space bar once. Then you can retype or correct the line using the left/right arrow keys.

CHAPTER 5

PLUG-INS AND ADD-ONS

The capabilities of your Laser 3000 can be expanded by attaching a number of peripherals to it. The following diagram shows what is possible.

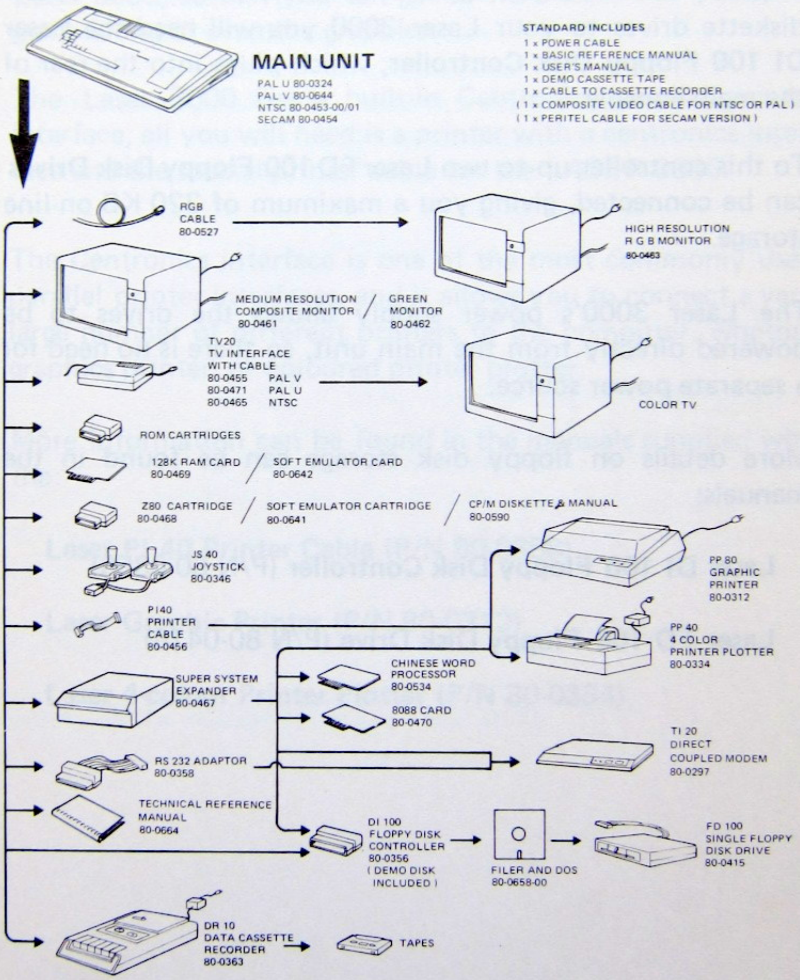


Figure 4.1 Laser 3000 System Diagram

DISKETTE DRIVES AND DISK CONTROLLER

Diskettes or "floppies" are the most popular form of program and data storage for microcomputers. To attach diskette drives to your Laser 3000 you will need the **Laser DI 100 Floppy Disk Controller**, which plugs into the rear of the main console.

To this controller, up to two **Laser FD 100 Floppy Disk Drives** can be connected, giving you a maximum of 320 KB on-line storage.

The Laser 3000's power supply allows the drives to be powered directly from the main unit, so there is no need for a separate power source.

More details on floppy disk storage can be found in the manuals:

Laser DI 100 Floppy Disk Controller (P/N 80-0356)

Laser FD 100 Floppy Disk Drive (P/N 80-0415)

PRINTER CABLE AND PRINTERS

At some stage you will want to connect a printer to your Laser 3000, so that you can get a "hard copy" of your program listing or even the graphics.

The Laser 3000 has a built-in Centronics parallel printer interface, all you will need is a printer with a centronics interface and the special printer cable for the LASER 3000.

The Centronics interface is one of the most commonly used parallel printer interfaces, and it allows you to connect a very large number of different printers to the computer, whether graphics printer, or coloured printer plotter.

More information can be found in the manuals supplied with the:

Laser PI 40 Printer Cable (P/N 80-0358)

Laser Graphic Printer (P/N 80-0312)

Laser 4-colour Printer Plotter (P/N 80-0334)

RS 232 ADAPTER

The RS232 interface is a widely used standard for connecting computer system components, especially for serial communication. The RS232 adapter allows you to connect the LASER 3000 to data terminal equipment such as CRT's and serial printers or to data communication equipment such as modems or network data concentrators.

The communication speed of the RS232 adaptor can be set to eight values between 110 and 9600 bauds.

More detail of the RS 232 Adapter in the manual for the:

Laser 3000 RS 232 Adapter (P/N 80-0358)

THE CARTRIDGE SLOT

On the right hand side of the console is a Cartridge slot. This connects directly to the Laser 3000's **System Bus.**

Hardware and solid state software modules can be attached to the Laser 3000 at this slot, expanding its capability greatly. The options are described below.

THE Z-80 CARTRIDGE

Plugging in the Z-80 cartridge not only add a second processor to the LASER 3000 (A Z80A, by the way. Your LASER 3000 has a 6502A microprocessor, if you still don't know, that is.), it also grant you access to a vast variety of software available under the CP/M® operating system, Such as word processors, data base packages, software development tools etc.

Further details available with the:

Laser 3000 Z-80 Cartridge (P/N 80-0468)

THE SOFT EMULATOR CARTRIDGE

If your version of LASER 3000 does not have the soft emulator card installed, it might be a good idea for you to get this cartridge. Especially if you want to gain a further benefit from the APPLE®][Software libary

LASER SOFT EMULATOR CARTRIDGE
(P/N 80-0041)

THE 8088 16-BIT MICROPROCESSOR CARTRIDGE

Like the Z-80 cartridge, this adds a second processor to the LASER 3000, only this time a 16-bit 8088! The software support for the 8088 is no less than the Z-80. We are pretty sure you will like the MS-DOS® and the CP/M®-86 operating system.

LASER 8088 CARTRIDGE (P/N 80-0470)

ROM CARTRIDGES

For those of you who do not have a floppy disk drive, we have a large library of plug-in software available in the form of ROM cartridges. Including application programs, games etc. All you will have to do is plug in the cartridge and then switch on.

More data on this in the manual with your:

Laser 3000 Solid State Application Cartridges

THE JOYSTICKS

A must for many computer games, a pair of joysticks can be attached to the Laser 3000 at the joystick port in the right hand side of the keyboard. How they work is fully described in the manual that comes with the:

Laser JS 40 Joystick (P/N 80-0346)

THE DATA CASSETTE

The data cassette is a low-cost alternative to floppy disk data and program storage. Although it is not as fast or as reliable as the floppy disk.

If you are going to use cassette tape for your storage, we recommend you use the data cassette rather than an ordinary cassette recorder.

The Laser DR 10 Data Cassette Recorder has been specially engineered to cope with some of the problems encountered when using an ordinary player. More information can be found in the manual with the:

Laser DR 10 Data Cassette Recorder (P/N 80-0363)

THE SUPER SYSTEM EXPANDER

With a system expander you can have more peripherals plugged in. All the standard Laser 3000 peripherals, cards, and ROM cartridges can be plugged into the system expander.

More details in the manual that is supplied with the:

Laser 3000 Super System Expander (P/N 80-0467)

CHAPTER 6

MEMORY ORGANISATION

MEMORY ORGANISATION

MEMORY CONFIGURATION

By some clever hardware design, the LASER 3000 can have a total of 262144 distinct memory locations (256K if you prefer). Trying to explain the memory organisation is no easy task, but here we go . . . .

Since the 6502 microprocessor can only address 65536 (or 64K) locations, we shall call this the System's LOGICAL MEMORY MAP. It can be thought of as 4 banks of memory, of 16K each, put together, as shown in figure 6.1

	MEMORY	USAGE
BANK 4	ROM ($C000–$FFFF)	SYSTEM KERNEL, BASIC INTERPRETER, INTERNAL I/0 LOCATIONS.
BANK 3	RAM ($8000–$BFFF)	FREE MEMORY GRAPHIC DISPLAY MEMORY
BANK 2	RAM ($4000–$7FFF)	GRAPHIC DISPLAY MEMORY FREE MEMORY
BANK 1	RAM ($0000–$3FFF)	LOW RESOLUTION GRAPHIC DISPLAY MEMORY TEXT DISPLAY MEMORY KERNEL VECTOR LOCATIONS KEYBOARD INPUT BUFFER SYSTEM STACK AND PROGRAMS

FIGURE 6.1 SYSTEM LOGICAL MEMORY MAP

On the other hand, the 256K memory (refered as the PHYSICAL MEMORY) is arranged as shown in figure 6.2. Each of the 16 blocks contains 16K of memory.

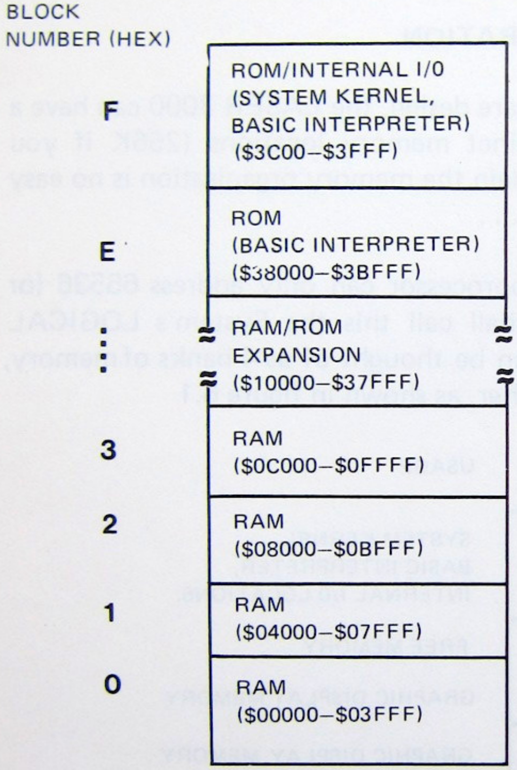


FIGURE 6.2 SYSTEM PHYSICAL MEMORY MAP.

Any of these block (0–F) in the physical memory map can be put in any bank of the system's logical memory map. For instance, after power up, block 0 is put into bank 1; block 1 into bank 2; block 3 into bank 3 and block F into bank 4. You can change this configuration by writing values to some special I/0 locations, although it might not help you that much by not putting the kernel (i.e. block F) in bank 4 – your LASER 3000 will simply go haywire. The TECHNICAL REFERENCE MANUAL will have a more detailed explanation of all these.

INTERNAL I/0 LOCATIONS

An area of memory locations of the LASER's memory map has been assigned to dedicated input and output functions. Their usage are somewhat intricate, full explanation is given in the TECHNICAL REFERENCE MANUAL (P/N 80-0664).

Table. 6.3 shows the allocation of INPUT/OUTPUT locations.

LOCATION (HEX)	FUNCTION
$ C000	read keyboard data
$ C010	clear keyboard storbe
$ C020	cassette output
$ C030	toggle speaker
$ C060	cassette input
$ C061	binary flag 1 input
$ C062	2
$ C063	3
$ C064	game paddle 1 input
$ C065	2
$ C066	3
$ C067	4
$ C068	write data to sound generator
$ C070	analog clear
$ C07C	write 1st memory window
$ C07D	2nd
$ C07E	3rd
$ C07F	4th
$ C090	write data to printer
$ C01C0	read printer acknowledge
$ C1C1	read printer busy
$ C1C2	read horizontal blanking
$ C1C3	read vertical blanking
$ C1C4	read 50/60 Hz switch status
$ C1C5	read high resolution switch

TABLE 6.3 INTERNAL I/0 LOCATIONS

SOFTWARE SWITCH LOCATIONS

A software switch is really an imaginary switch which can be throw into the 'on' or 'off' position through software control only. There are a number of software switches in the LASER 3000. Each of these switches has been allocated to a specific address, table 6.4 is a list of these locations. For more detailed information refer to the TECHNICAL REFERENCE MANUAL (P/N 80-0664)

LOCATION (HEX)	FUNCTION
	SETTING COLOURS
$ C008	border colour to black
$ C009	red
$ C00A	green
$ C00B	yellow
$ C00C	blue
$ C00D	magenta
$ C00E	cyan
$ C00F	white
$ C018	background colour to black
$ C019	red
$ C01A	green
$ C01B	yellow
$ C01C	blue
$ C01D	magenta
$ C01E	cyan
$ C01F	white
$ C028	background to black and foreground to white
$ C029	Foreground colour to green
$ C02A	yellow
$ C02B	blue
$ C02C	magenta
$ C02D	cyan
$ C02E	white
	SELECTING DISPLAY MODES
$ C04C	low resolution mode
$ C04D	RGB mode
$ C04E	double resolution mode
$ C04F	80-col text mode
$ C050	graphics mode
$ C051	text mode
$ C052	pure text or graphics mode
$ C053	mixed text and graphics mode
$ C054	primary page
$ C055	secondary page
$ C056	disable RGB and double resolution selection
$ C057	enable RGB and double resolution selection

TABLE 6.4 SOFTWARE SWITCH LOCATIONS

To select the switch, all you need to do is 'reference' the appropriate location. By 'reference', we mean you can either perform a read or write on that location. For example.
enter the kernel by typing SHIFT and F1 (alternatively, if you are the conservative type, enter CALL-151 and RETURN). Then type:

C009 RETURN

then you should have you screen framed in a red colour. Try the others locations and see what happens.

CHAPTER 7

THE SYSTEM KERNEL

THE SYSTEM KERNEL

KERNEL OVERVIEW

The kernel (or sometimes referred as the monitor) is a program inplementing the fundamental set of commands required to operate a computer system. The LASER 300 system kernel provides you with a number of useful features, such as examining memory locations, changing their contents, moving or comparing block of data quickly etc.

IN AND OUT OF THE KERNEL

To enter the System kernel, type.

CALL –151 RETURN

When you see the basic prompt sign (i.e.]) or simply type SHIFT F1 then it should bring you to the kernel. The kernel's prompt (an asterisk [*]) should appear, and you can enter any of the kernel commands listed in the next section. The commands line you type in can be as long as 255 characters, so you can put several commands on a single line, as long as you separate them with spaces. The compuer will take action only after you hit RETURN. The kernel takes single character commands, and it can only recognise hexadecimal numbers.

If you've got fed up with the kernel, typing

CTRL C and then RETURN

or pressing the RESET button on the back of the console will bring you back to BASIC (with D.O.S. you can use 3DØG RETURN).

TALKING IN HEXADECIMAL

Since the kernel only understands a rather primitive dialect called Hexadecimal. So we must be patient and speak Hexadecimal when we talk to the kernel.

EXAMINING AND CHANGING MEMORY CONTENTS

To examine a memory location, simply type in the address and hit [RETURN] , the computer will report to you the value contained in that location (all in hexadecimal, of course). Hitting [RETURN] a second time will cause the kernel to respond with one line of memory dump, i.e. displaying the contents of the location following the last opened location to the next eight-location 'cut' (or the next sixteen-location 'cut' in 80-column mode). Thus to examine a range of memory, simply type in the address of the starting location, and press [RETURN] as many times as necessary.

For example:

```
*D000 ([RETURN])
D000=6F
* ([RETURN])
D001=D8 65 D7 F8 DC 94 D9
```

To change the content of a mamory location, type the address of the location followed by a colon and a hexadecimal value.

*6000 (RETURN)
*6000=FF
*6000:13
*6000 (RETURN)
*6000=13

To change consecutive memory locations, you only need to type the initial adddress and a colon, and then all values for each consecutive memory separated by spaces. A total of eighty-five location can be changed at a time this way.

MOVING AND COMPARING MEMORY

The M(ove) command allows you to move a block of memory from one place to another in memory.

The V(erify) command compares the contents of 2 blocks of memory, and reports the addresses and values of all discrepant values.

The format of the two commande are the same:

destination < *start* . *end* M

will make a copy of the indicated range to the specified destination.

destination < *start* . *end* V

will compare the range specified with the range beginning at the destination address.

USING THE CASSETTE TAPE

There are two special commands which allow you to save memory contents onto cassette tape and retrieve it again. To write a range of memory to tape, you will need to specify the starting and ending addresses, followed by the letter W (for WRITE).

start . *end* W RETURN

To retrieve data stored on tape with the W(rite) command, use the R(ead) commant. Again, specify the starting and ending addresses, then the letter R (for READ)

start . *end* R RETURN

RUNNING MACHINE LANGUAGE PROGRAMS

The G(0) command can be used to transfer control to the machine language program at an address you specified.

address G RETURN

for example typing E000G RETURN will take you to BASIC.

More information on the kernel program and commands is available in the TECHNICAL REFERENCE MANUAL.

APPENDIX I

VIDEO SCREEN MAPPING

VIDEO SCREEN MAPPING

40/80-COLUMN TEXT DISPLAY

The memory area allocated for the primary 40-column text page starts at location number 1024 (=$400), and extends to location number 2047 (=$7FF). The second screen begins at location 2048 (=$800) and extends to location 3071 (=$BFF). Each of these pages occupies 1024 bytes of memory. While in 80-column mode, the primary page takes up the memory between physical location 4096 (=$01000) and physical location 6143 (=$017FF). The second 80-column text page starts at location 6144 (=$01800) and ends at location 8191 (=01FFF).

Figures A1.1, A1.2 and A1.3 are the maps of the LASER'S display in 40-column and 80-column text mode, each character position on the screen are shown in both decimal and hexadecimal form.

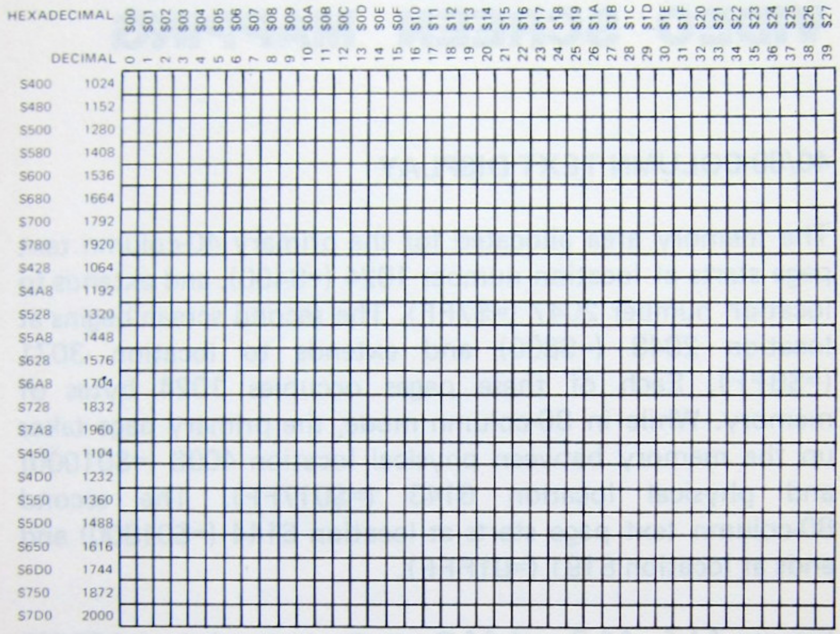


Fig A1.1 40 column text display memory map.

COL. No.		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39
HEXADECIMAL		$00	$01	$02	$03	$04	$05	$06	$07	$08	$09	$0A	$0B	$0C	$0D	$0E	$0F	$10	$11	$12	$13	$14	$15	$16	$17	$18	$19	$1A	$1B	$1C	$1D	$1E	$1F	$20	$21	$22	$23	$24	$25	$26	$27
DECIMAL		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39
4096	$1000																																								
4224	$1080																																								
4352	$1100																																								
4480	$1180																																								
4608	$1200																																								
4736	$1280																																								
4864	$1300																																								
4992	$1380																																								
4136	$1028																																								
4264	$10A8																																								
4392	$1128																																								
4520	$11A8																																								
4648	$1228																																								
4776	$12A8																																								
4904	$1328																																								
5032	$13A8																																								
4176	$1050																																								
4304	$10D0																																								
4432	$1150																																								
4560	$11D0																																								
4688	$1250																																								
4816	$12D0																																								
4944	$1350																																								
5072	$13D0																																								

Fig A1.2 80 column text display memory map (left hand side)

	COL. No.	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
	HEXADECIMAL	$00	$01	$02	$03	$04	$05	$06	$07	$08	$09	$0A	$0B	$0C	$0D	$0E	$0F	$10	$11	$12	$13	$14	$15	$16	$17	$18	$19	$1A	$1B	$1C	$1D	$1E	$1F	$20	$21	$22	$23	$24	$25	$26	$27
	DECIMAL	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39
5120	$1400																																								
5248	$1480																																								
5376	$1500																																								
5504	$1580																																								
5632	$1600																																								
5760	$1680																																								
5888	$1700																																								
6016	$1780																																								
5160	$1428																																								
5288	$14A8																																								
5416	$1528																																								
5544	$15A8																																								
5672	$1628																																								
5800	$16A8																																								
5928	$1728																																								
6056	$17A8																																								
5200	$1450																																								
5328	$14D0																																								
4656	$1550																																								
5584	$15D0																																								
5712	$1650																																								
5840	$16D0																																								
5968	$1750																																								
6069	$17D0																																								

Fig A1.3 80 column text display memory may (right hand side)

LOW-RESULUTION GRAPHIC DISLPLAY

The 2 pages in the LASER'S low resolution graphics mode occupy a total of 16K byte memory area. The primary page begins at physical memory location number 8192 (=$02000) and extends up to location number 16383 (=$03FFF). The secondary page starts at location 16384 (=$04000) and finishes at location 24575 (=$05FFF).

Figure A1.4 shows the LASER'S display screen in low resolution graphics mode with the memory addresses of each line on the screen.

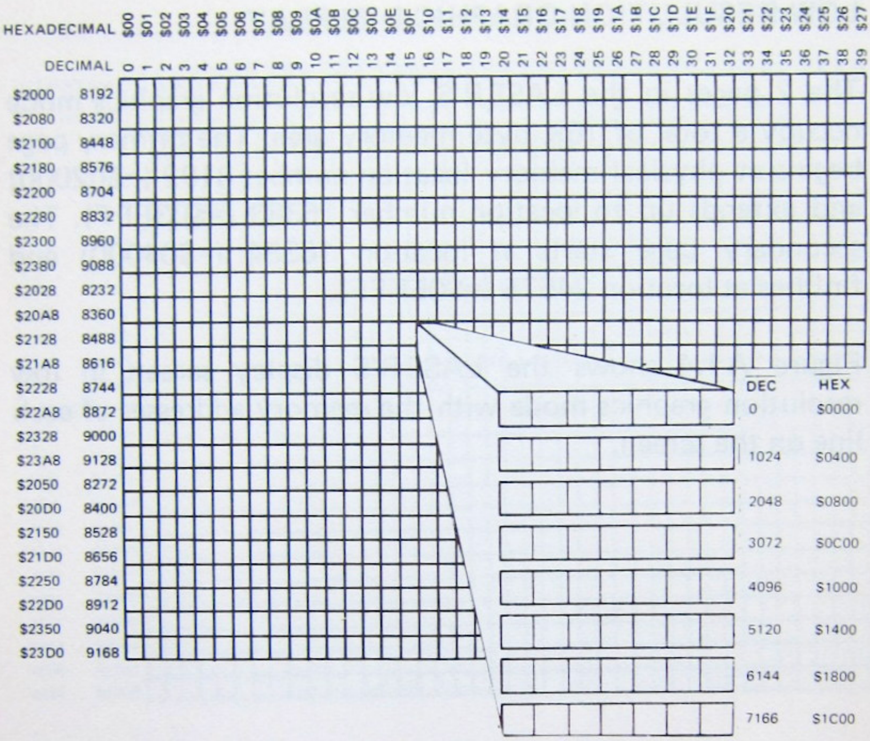


Fig A1.4 low-resolution graphics display memory map

BIT-IMAGE GRAPHICS DISPLAY

The Bit-Image Graphics mode takes it data from an area of 24576-byte (24K) memory. This area of memory is further, divided into 3 sections, each of these section represents one of the primary colours (i.e. red, green and blue), the secondary colours can be obtained by mixing these three colours. The primary page of the Bit-Image Graphics mode starts at physical memory location 16384 (=$04000) and extends to location 40959 (=$09FFF), the secondary follows immediately from location 40960 (=$0A000) to location 65535 (=$0FFFF).

Figure A1.5 shows the display screen map in Bit-Image Graphics mode, with the memory address of the Red, Green and Blue planes, as well as location number of each line on the screen.

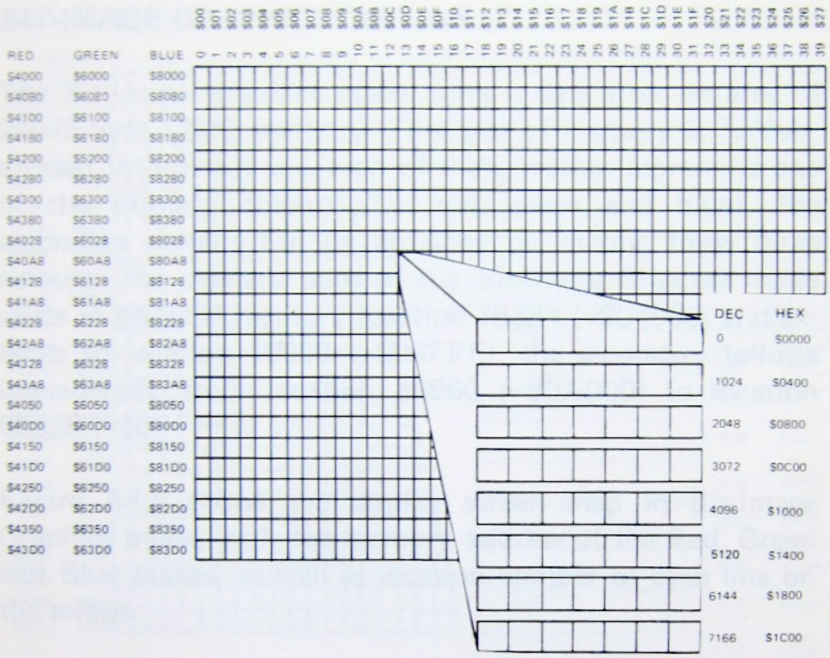


Fig A1.5 Bit image graphics display memory map.

DOUBLE RESOLUTION GRAPHICS DISPLAY

The Double Resolution Graphics mode can be thought of as being 2 Low Resolution pages put side by side on the display screen. The primary and secondary page in this graphic mode occupies a 16384-byte (16K) memory area each.

The primary page starts at physical memory location 16384 (=$04000) and extends up to location 32767 (=$07FFF). The secondary page begins at location 32768 (=$08000) and ends at 49151 (=$0BFFF).

Figure A1.6 and A1.7 show the display screen map in Double Resolution Graphics mode, with the memory addresses of each line on the screen.

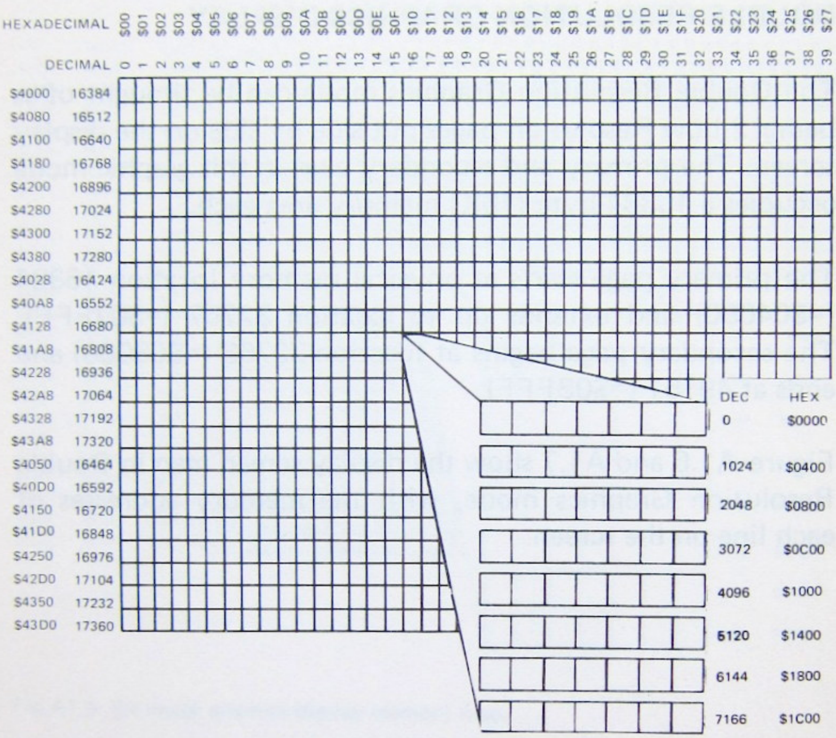


Fig A1.6 Double Resolution graphics display memory map (left hand side).

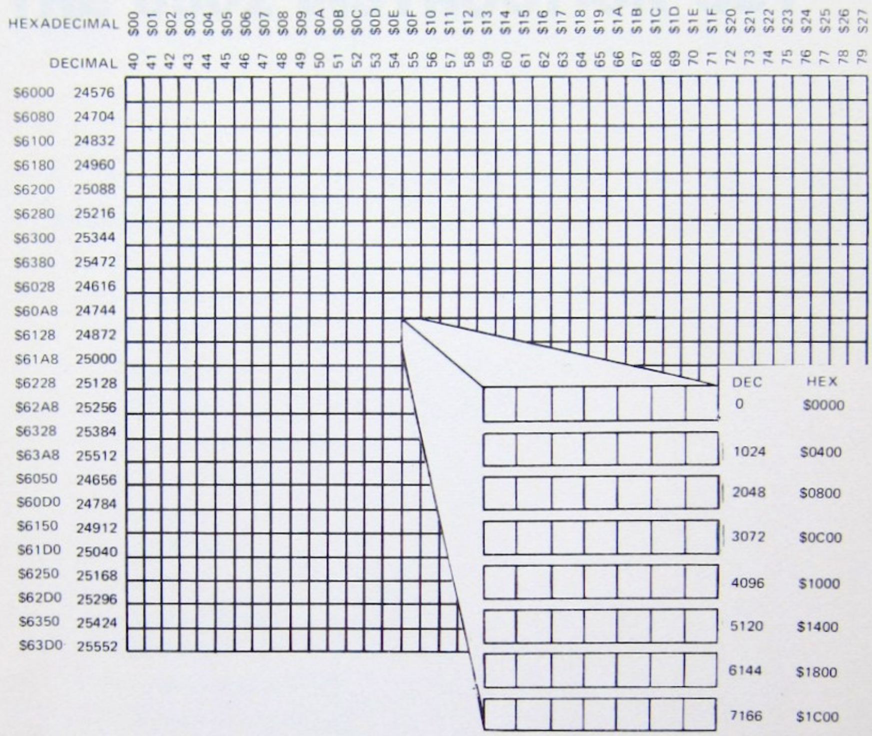


Fig A1.7 Double Resolution graphics display memory (right hand side)

APPENDIX II

THE 6502 INSTRUCTION SET

THE 6502 INSTRUCTION SET

THE FOLLOWING NOTATION APPLIES TO THIS SUMMARY:

A	Accumulator
X, Y	Index Registers
M	Memory
$\overline{C}$	Borrow
P	Processor Status Register
S	Stack Pointer
✓	Change
—	No Change
+	Add
∧	Logical AND
-	Subtract
⊻	Logical Exclusive Or
↑	Transfer From Stack
↓	Transfer To Stack
→	Transfer To
←	Transfer To
V	Logical OR
PC	Program Counter
PCH	Program Counter High
PCL	Program Counter Low
OPER	Operand
#	Immediate Addressing Mode

FIGURE A2.1 ASL-SHIFT LEFT ONE BIT OPERATION

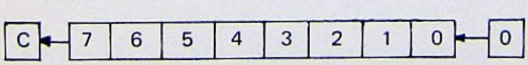


FIGURE A2.2 ROTATE ONE BIT LEFT (MEMORY OR ACCUMULATOR)

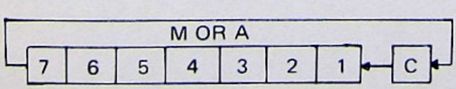
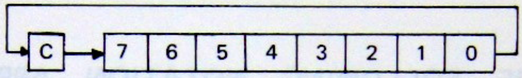


FIGURE A2.3 ROTATE ONE BIT RIGHT (MEMORY OR ACCUMULATOR)



NOTE 1: BIT – TEST BITS

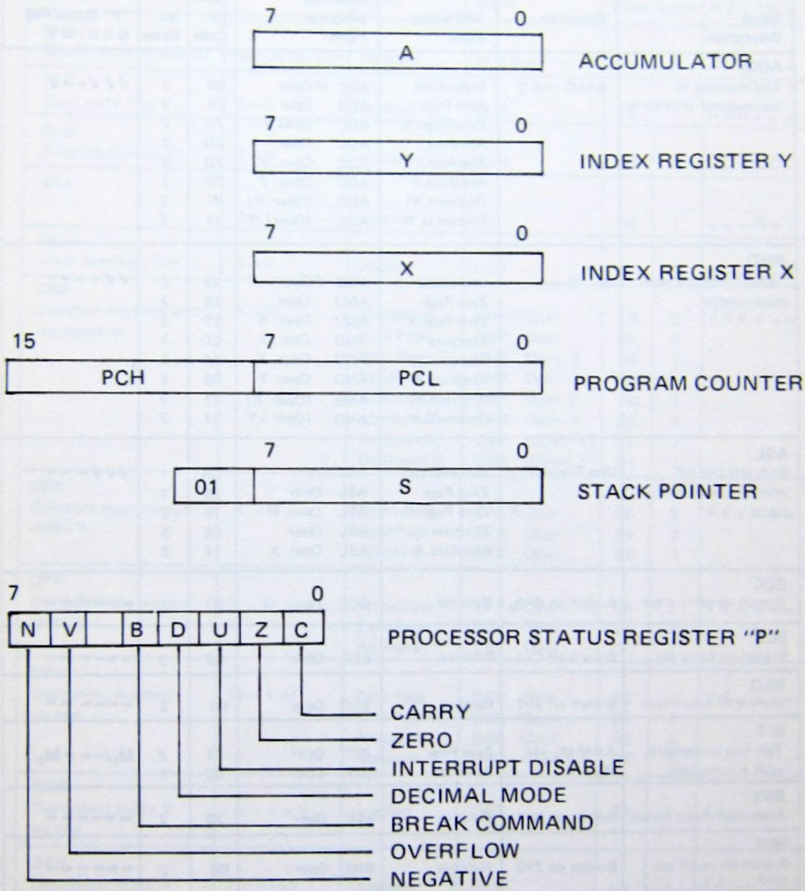
Bit 6 and 7 are transferred to the status register. If the result of A ∧ M is zero then Z=1, otherwise Z=0.

6502 MICROPROCESSOR INSTRUCTIONS

ADC	Add memory to Accumulator with Carry
AND	"AND" Memory with Accumulator
ASL	Shift Left One Bit (Memory or Accumulator)
BCC	Branch on Carry Clear
BCS	Branch on Carry Set
BEQ	Branch on Result Zero
BIT	Test Bits in Memory with Accumulator
BMI	Branch on Result Minus
BNE	Branch on Result not Zero
BPL	Branch on Result Plus
BRK	Force Break
BVC	Branch on Overflow Clear
BVS	Branch on Overflow Set
CLC	Clear Carry Flag
CLD	Clear Decimal Mode
CLI	Clear Interrupt Disable Bit
CLV	Clear Overflow Flag
CMP	Compare Memory and Accumulator
CPX	Compare memory and Index X
CPY	Compare Memory and Index Y
DEC	Decrement Memory by One
DEX	Decrement Index X by One
DEY	Decrement Index Y by One
EOR	"Exclusive-Or" Memory with Accumulator
INC	Increment Memory by One
INX	Increment Index X by One
INY	Increment Index Y by One
JMP	Jump to New Location
JSR	Jump to New Location Saving Return Address

LDA	Load Accumulator with Memory
LDX	Load Index X with Memory
LDY	Load Index Y with Memory
LSR	Shift Right one Bit Memory or Accumulator
NOP	No Operation
ORA	"OR" Memory with Accumulator
PHA	Push Accumulator on Stack
PHP	Push Processor Status on Stack
PLA	Pull Accumulator from Stack
PLP	Pull Processor Status from Stack
ROL	Rotate One Bit Left (Memory or Accumulator)
ROR	Rotate One Bit Right (Memory or Accumulator
RTI	Return from Interrupt
RTS	Return from Subroutine
SBC	Subtract Memory from Accumulator with Borrow
SEC	Set Carry Flag
SED	Set Decimal Mode
SEI	Set Interrupt Disable Status
STA	Store Accumulator in Memory
STX	Store Index X in Memory
STY	Store Index Y in Memory
TAX	Transfer Accumulator to Index X
TAY	Transfer Accumulator to Index Y
TSX	Transfer Stack Pointer to Index X
TXA	Transfer Index X to Accumulator
TXS	Transfer Index X to Stack Pointer
TYA	Transfer Index Y to Accumulator

PROGRAMMING MODEL



INSTRUSTION CODES

Name Description	Operation	Addressing Mode	Assembly Language Form	HEX OP Code	No. Bytes	"P" Status Reg. N Z C I D V
ADC Add memory to accumulator with carry	A·M·C → A.C	Immediate	ADC #Oper	69	2	√√√--√
		Zero Page	ADC Oper	65	2	
		Zero Page. X	ADC Oper X	75	2	
		Absolute	ADC Oper	6D	3	
		Absolute.X	ADC Oper. X	7D	3	
		Absolute. Y	ADC Oper. Y	79	3	
		(Indirect.X)	ADC (Oper. X)	61	2	
		(Indirect), Y	ADC (Oper). Y	71	2	
AND "AND" memory with accumulator	A M → A	Immediate	AND #Oper	29	2	√√----
		Zero Page	AND Oper	25	2	
		Zero Page.X	AND Oper. X	35	2	
		Absolute	AND Oper. X	2D	3	
		Absolute.X	AND Oper. X	3D	3	
		Absolute Y	AND Oper. Y	39	3	
		(Indirect.X)	AND (Oper. X)	21	2	
		(Indirect),Y	AND (Oper.) Y	31	2	
ASL Shift left one bit (Memory or Accumulator)	(See Figure 1)	Accumulator	ASL A	0A	1	√√√---
		Zero Page	ASL Oper	06	2	
		Zero Page X	ASL Oper. X	16	2	
		Zbsolute	ASL Oper	0E	3	
		Absolute X	ASL Oper. X	1E	3	
BCC Branch on carry clear	Branch on C=0	Relative	BCC Oper	90	2	------
BCS Branch on carry set	Branch on C=1	Relative	BCS Oper	B0	2	------
BEQ Branch on result zero	Branch on Z=1	Relative	BEQ Oper	F0	2	------
BIT Test bits in memory with accumulator	A∨M $M_7 \to N$. $M_6 \to V$	Zero Page	BIT[1] Oper	24	2	M_7√---M_6
		Absolute	BIT[1] Oper	2C	3	
BMI Branch on result minus	Branch on N=1	Relative	BMI Oper	30	2	------
BNE Branch on result not zero	Branch on Z=0	Relative	BNE Oper	D0	2	------
BPL Branch on result plus	Branch on N=0	Relative	BPL Oper	10	2	------
BRK Force Break	Forced Interrupt PC+2 ↓P↓	Implied	BRK[2]	00	1	---1--
BVC Branch on overflow clear	Branch on V=0	Relative	BVC Oper	50	1	------

Note 1: Bits 6 and 7 are transferred to the status register. If the result of A AND M is zero. then Z = 1; otherwise Z = 0.
Note 2: A BRK command cannot be masked by setting I.

Name Description	Operation	Addressing Mode	Assembly Language Form	HEX OP Code	No. Bytes	"P" Status Reg. N Z C I D V
BVS Branch on overflow set	Branch on V=1	Relative	BVS Oper	70	2	------
CLC Clear carry flag	0→C	Implied	CLD	18	1	--0---
CLD Clear decimal mode	0→D	Implied	CLD	D8	1	----0-
CLI	0→1	Implied	CLI	58	1	---0--
CLV Clear overflow flag	0→V	Implied	CLV	B8	1	-----0
CMP Compare memory and accumulator	A – M	Immediate Zero Page Zero Page. X Absolute Absolute.X Absolute.Y (Indirect.X) (Indirect).Y	CMP # Oper CMP Oper CMP Oper. X CMP Oper CMP Oper. X CMP Oper. Y CMP (Oper. X) CMP (Oper).Y	C9 C5 D5 CD DD D9 C1 D1	2 2 2 3 3 3 2 2	√√√---
CPX Compare memory and index X	X – M	Immediate Zero Page Absolute	CPX # Oper CPX Oper CPX Oper	E0 E4 EC	2 2 3	√√√---
CPY Compare memory and index Y	Y – M	Immediate Zero Page Absolute	CPY # Oper CPY Oper CPY Oper	C0 C4 CC	2 2 3	√√√---
DEC Decrement memory by one	M – 1→M	Zero Page Zero Page X Absolute Absolute. X	DEC Oper DEC Oper. X DEC Oper DEC Oper. X	C6 D6 CE DE	2 2 3 3	√√----
DEX Decrement index Y by one	X – 1→X	Implied	DEX	CA	1	√√----
DEY Decrement index Y by one	Y – 1→Y	Implied	DEY	88	1	√√----

Name Description	Operation	Addressing Mode	Assembly Language Form	HEX OP Code	No. Bytes	"P" Status Reg. N Z C I D V
EOR "Exclusive-Or" memory with accumulator	A V M → A	Immediate	EOR #Oper	49	2	√√----
		Zero Page	EOR Oper	45	2	
		Zero Page, X	EOR Oper, X	55	2	
		Absolute	EOR Oper	4D	3	
		Absolute, X	EOR Oper, X	5D	3	
		Absolute, Y	EOR Oper,Y	59	3	
		(Indirect, X)	EOR (Oper, X)	41	2	
		(Indirect), Y	EOR (Oper),Y	51	2	
INC Increment memory by one	M + 1 → M	Zero Page	INC Oper	E6	2	√√----
		Zero Page, X	INC Oper, X	F6	2	
		Absolute	INC Oper	EE	3	
		Absolute, X	INC Oper, X	FE	3	
INX Increment index X by one	X + 1 → X	Implied	INX	E8	1	√√----
INY Increment index X by one	Y + 1 → Y	Implied	INY	C8	1	√√----
JMP Jump to new location	(PC+1) → PCL (PC+2) → PCH	Absolute	JMP Oper	4C	3	------
		Indirect	JMP (Oper)	6C	3	
JSR Jump to new location saving return address	PC+2↓ (PC+1) → PCL (PC+2) → PCH	Absolute	JSR Oper	20	3	------
LDA Load accumulator with memory	M → A	Immediate	LDA#Oper	A9	2	√√----
		Zero Page	LDA Oper	A5	2	
		Zero Page, X	LDA Oper, X	B5	2	
		Absolute	LDA Oper	AD	3	
		Absolute, X	LDA Oper, X	BD	3	
		Absolute, Y	LDA Oper,Y	B9	3	
		(Indirect, X)	LDA (Oper, X)	A1	2	
		(Indirect), Y	LDA (Oper), Y	B1	2	
LDX Load index X with memory	M → X	Immediate	LDX#Oper	A2	2	√√----
		Zero Page	LDX Oper	A6	2	
		Zero Page, Y	LDX Oper, Y	B6	2	
		Absolute	LDX Oper	AE	3	
		Absolute, Y	LDX Oper, Y	BE	3	
LDY Load index Y with memory	M → Y	Immediate	LDY#Oper	A0	2	√√----
		Zero Page	LDY Oper	A4	2	
		Zero Page, X	LDY Oper, X	B4	2	
		Absolute	LDY Oper	AC	3	
		Absolute, X	LDY Oper, X	BC	3	

Name Description	Operation	Addressing Mode	Assembly Language Form	HEX OP Code	No. Bytes	"P" Status Reg. N Z C I D V
LSR Shift right one bit (memory or accumulator)	(See Figure 1)	Accumulator	LSR A	4A	1	0 √ √ – – –
		Zero Page	LSR Oper	46	2	
		Zero Page, X	LSR Oper, X	56	2	
		Absolute	LSR Oper	4E	3	
		Absolute, X	LSR Oper, X	5E	3	
NOP No operation	No Operation	Implied	NOP	EA	1	– – – – – –
ORA "OR" memory with accumulator	A V M → A	Immediate	ORA #Oper	09	2	√ √ – – – –
		Zero Page	ORA Oper	05	2	
		Zero Page, X	ORA Oper,X	15	2	
		Absolute	ORA Oper	0D	3	
		Absolute, X	ORA Oper,X	1D	3	
		Absolute, Y	ORA Oper,Y	19	3	
		(Indirect,X)	ORA (Oper,X)	01	2	
		(Indirect), Y	ORA (Oper), Y	11	2	
PHA Push accumulator on stack	A ↓	Implied	PHA	48	1	– – – – – –
PHP Push processor status on stack	P ↓	Implied	PHP	08	1	– – – – – –
PLA Pull processor status from stack	A ↑	Implied	PLA	68	1	√ √ – – – –
PLP Pull processor status from stack	P ↑	Implied	PLP	28	1	From Stack
ROL Rotate one bit left (memory or accumulator)	(See Figure 2)	Accumulator	ROL A	2A	1	√ √ √ – – –
		Zero Page	ROL Oper	26	2	
		Zero Page, X	ROL Oper,X	36	2	
		Absolute	ROL Oper	2E	3	
		Absolute,X	ROL Oper,X	3E	3	
ROR Rotate one bit right (memory or accumulator)	(See Figure 3)	Accumulator	ROR A	6A	1	√ √ √ – – –
		Zero Page	ROR Oper.	66	2	
		Zero Page,X	ROR Oper, X	76	2	
		Absolute	ROR Oper	6E	3	
		Absolute,X	ROR Oper,X	7E	3	

Name Description	Operation	Addressing Mode	Assembly Language Form	HEX OP Code	No. Bytes	"P" Status Reg. N Z C I D V
RTI Return from interrupt	P↑ PC↑	Implied	RTI	40	1	From Stack
RTS Return from subroutine	PC↑PC-1→PC	Implied	RTS	60	1	------
SBC Subtract memory from accumulator with borrow	A - M - $\overline{C}$ → A	Immediate	SBC #Oper	E9	2	√ √ √ - - √
		Zero Page	SBC Oper	E5	2	
		Zero Page.X	SBC Oper. X	F5	2	
		Absolute	SBC Oper	ED	3	
		Absolute.X	SBC Oper.X	FD	3	
		Absolute.Y	SBC Oper. Y	F9	3	
		(Indirect.X)	SBC (Oper. X)	F1	2	
		(Indirect). Y	SBC (Oper). Y	F1	2	
SEC Set carry flag	1→C	Implied	SEC	38	1	- - 1 - - -
SED Set decimal mode	1→D	Implied	SED	F8	1	- - - - 1 -
SEI Set interrupt disable status	1→I	Implied	SEI	78	1	- - - 1 - -
STA Store accumulator in memory	A→M	Zero Page	STA Oper	85	2	------
		Zero Page.X	STA Oper. X	95	2	
		Absolute	STA Oper	8D	3	
		Absolute. X	STA Oper.X	9D	3	
		Absolute. Y	STA Oper. Y	99	3	
		(Indirect.X)	STA (Oper.X)	81	2	
		(Indirect). Y	STA (Oper). Y	91	2	
STX Store index X in memory	X→M	Zero Page	STX Oper	86	2	------
		Zero Page.Y	STX Oper. Y	96	2	
		Absolute	STX Oper	8E	3	
STY Store index Y in memory	Y→M	Zero Page	STY Oper	84	2	------
		Zero Page.X	STY Oper. X	94	2	
		Absolute	STY Oper	8C	3	
TAX Transfer accumulator to index X	A→X	Implied	TAX	AA	1	√ √ - - - -
TAY Transfer accumulator to index Y	A→Y	Implied	TAY	A8	1	√ √ - - - -
TSX Transfer stack pointer to index X	S→X	Implied	TSX	BA	1	√ √ - - - -

Name Description	Operation	Addressing Mode	Assembly Language Form	HEX OP Code	No. Bytes	"P" Status Reg. N Z C I D V
TXA Transfer Index X to accumulator	X→A	Implied	TXA	8A	1	√ √ - - - -
TXS Transfer Index X to stack pointer	X→S	Implied	TXS	9A	1	- - - - - -
TYA Transfer Index Y to accumulator	Y→A	Implied	TYA	98	1	√ √ - - - -

®APPLE and ®APPLESOFT are registered trademark of Apple Computer Inc.
®CP/M is a registered trademark of Digital Research Inc.
®MS-DOS is a registered trademark of Microsoft Inc.

LASER™
3000
PERSONAL COMPUTER

91-0172-08