LASER 128 SERIES

APPLE® IIc / IIe COMPATIBLE COMPUTER

Technical Reference Manual

VTECH

The following message is applicable to FCC Class B version units only :

Note: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15, of FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures : ,

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/television technician for help.

You may find the following booklet prepared by the Federal Communications Commission helpful: "How to Identify and Resolve Radio-TV Interference Problems" This booklet is available from the U.S. Government Printing Office, Washington, DC20402, Stock No. 004-000-00345-4.

To ensure that the use of this product does not contribute to interference, it is necessary to use shielded I/O cables.

TABLE OF CONTENTS

APPENDIX

CHAPTER 1
INTRODUCTION

1. INTRODUCTION

This manual is primarily written for those users who need to know the software and hardware details of the computer in order to utilize its features more efficiently. These include application programmers, hardware designers as well as hobbyists. It provides specific information on the internal workings of the computer and is intended to be a technical supplement to the user's manual that comes with your computer.

1.1 Organisation of manual

This manual can be conceptually divided into two sections.

Chapters 1 to 10 describe the functional and software aspects of the computer in a programmer's point of view. The organisation and management of the memory and I/O devices are described in detail. After going through these chapters, the reader should be able to access the resources on the computer directly, by-passing the built-in firmware. (Note: For compatibility reasons, access to these resources should be made through the built-in firmware whenever possible.)

Chapter 11 describes the hardware aspect of the computer. The entire circuit is divided into functional blocks and investigated one by one in depth.

This chapter is particularly useful for those service personnel who need to know the function of each component, the structure and organisation of the system buses, the general signal flow and timing waveforms of the various signals.

Besides, it also contains valuable information on the function and electrical characteristic of the signals on the various expansion connectors on the computer which is required by the peripheral hardware designers.

The appendices at the end of this manual provides supplementary information to the reader. These include schematics, PCB and component layouts, parts lists, gate-array pin assignments and data sheets for some of the major components used in the computer.

1.2 Other sources of information

The materials provided in this manual is adhered to those technically inclined users. It assumes that the reader has some background knowledge on the basic structure and general operating principles of microcomputer systems and common peripheral devices.

The descriptions throughout this manual are clear, simple and easy to understand. Technical terms are avoided as far as possible and where they are used, they are explained clearly.

However if the reader happens to be a first-time computer user, it is suggested that the reader should read some other reference books on relevant subjects, e.g. microprocessor system design, operating principles of peripheral devices such as disk drives and TV/monitors, assembly language programming etc.. This will help the reader in understanding the materials described in this manual more thoroughly.

This manual focuses on the internals of the computer. To obtain further information on the installation and general operations of the computer, the readers are referred to the user's manual.

CHAPTER 2
CENTRAL PROCESSING UNIT

2. CENTRAL PROCESSING UNIT

The computer uses 65C02, an enhanced version of its predecessor 6502, as its central processing unit. In the "LASER 128", 1 MHz version is used. In the "LASER 128 EX", 4 MHz version is used.

It has the following advantages compared with the NMOS 6502:

- Uses CMOS (Complementary Metal Oxide Semiconductor) technology to reduce power consumption and increase noise immunity.
- Completely static operation, no lower-limit on CPU clock frequency (can be d.c.).
- Enhanced instruction set with new instructions and addressing modes.

The 65C02 is a 8-bit microprocessor having 16 address lines to access a 64 K-byte address space. It has one 16-bit program counter, one 8-bit stack pointer, three 8-bit general purpose registers and one 8-bit processor status register. Figure 2-1 shows the programming model.

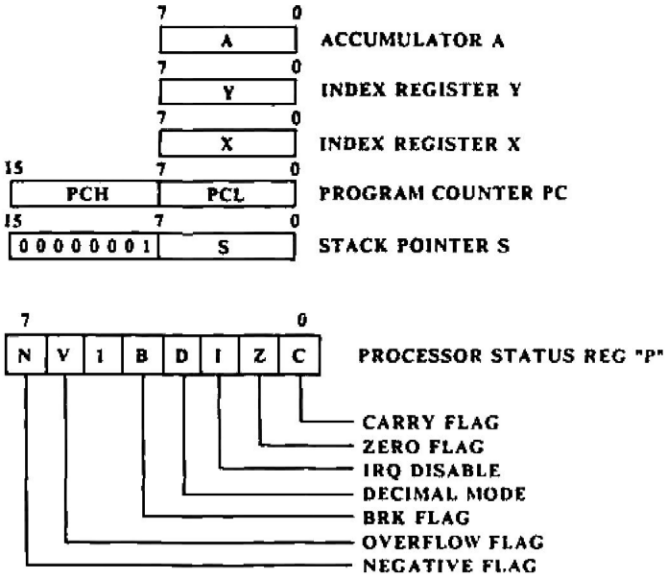


Fig. 2-1 Programming model of 65C02

The program counter "PC" is an 16-bit counter which keeps track of the address of the next instruction to be fetched and executed.

The stack pointer "S" is an 8-bit register which keeps track of the low-order eight bits of the address of the "top" of a 256-byte RAM stack maintained in $0100 to $01FF of the address space. The stack is primarily used for storing the subroutine call or interrupt return addresses. Use of the stack allows nested subroutine calls and multiple level interrupts.

The accumulator "A" is an 8-bit general purpose register which is often used for storing the result of arithmetic and logical operations. It also forms one of the input operands for many of these operations.

The index registers "X" and "Y" are 8-bit registers which are usually used with the indexed addressing mode of the microprocessor. Use of these registers facilitates the manipulation of tables maintained in main memory.

The processor status register "P" is an 8-bit register which holds the internal state of the microprocessor and status of some arithmetic and logical operations.

The above is only a brief introduction to the 65C02 CPU. For more information, please refer to the data sheet and technical documentations published by the manufacturers.

CHAPTER 3
MEMORY ORGANISATION

3. MEMORY ORGANISATION

The ROM and RAM memory system will be investigated in this chapter.

3.1 System address map

Figure 3-1 shows the system address map. The CPU address space can be divided into four areas:

a) ROM (Read Only Memory)
b) RAM (Random Access Memory)
c) I/O (Input / Output)
d) Hardware page

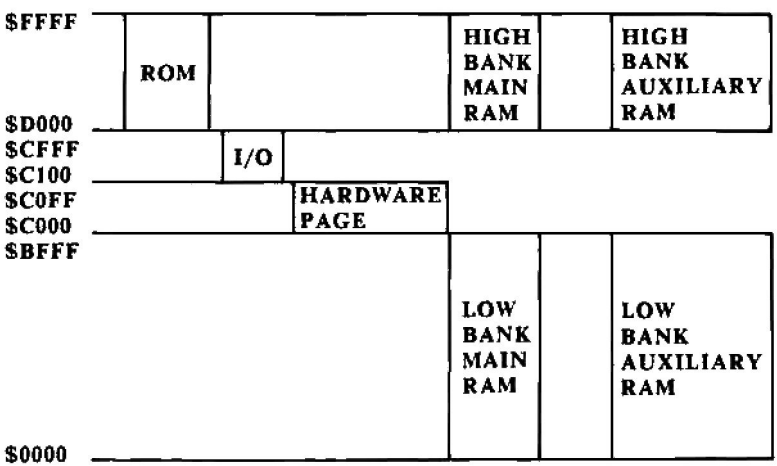


Fig. 3-1 System address map

The 65C02 has 16 address lines. It can only access 64K bytes of memory. In order to access more than 64K bytes of memory, the bank-switching technique is employed. There are bank registers to map the physical memory to the 64K-byte logical address space.

3.2 ROM

The ROM consists of three parts:

a) BASIC interpreter
b) System monitor
c) I/O drivers

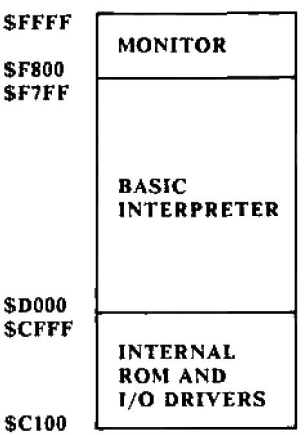


Fig. 3-2 ROM mapping

INTIOROM is a soft-switch which controls the I/O ROM bank configuration. Soft-switch is widely used in the computer. The logical state of the soft-switch is set by software. A summary of all soft-switches are listed in Table 3-5.

3.2.1 The BASIC interpreter

The address range is from $D000 to $F7FF. Details of the BASIC commands can be found in the user's manual.

3.2.2 The System Monitor

The monitor occupies $F800 to $FFFF. The basic I/O subroutines are in the monitor. You can enter the monitor from BASIC by typing CALL-151 and pressing RETURN.

3.2.3 I/O firmwares

3.2.3.1 I/O firmwares map

The mapping of the I/O firmwares are illustrated in Fig 3-3.

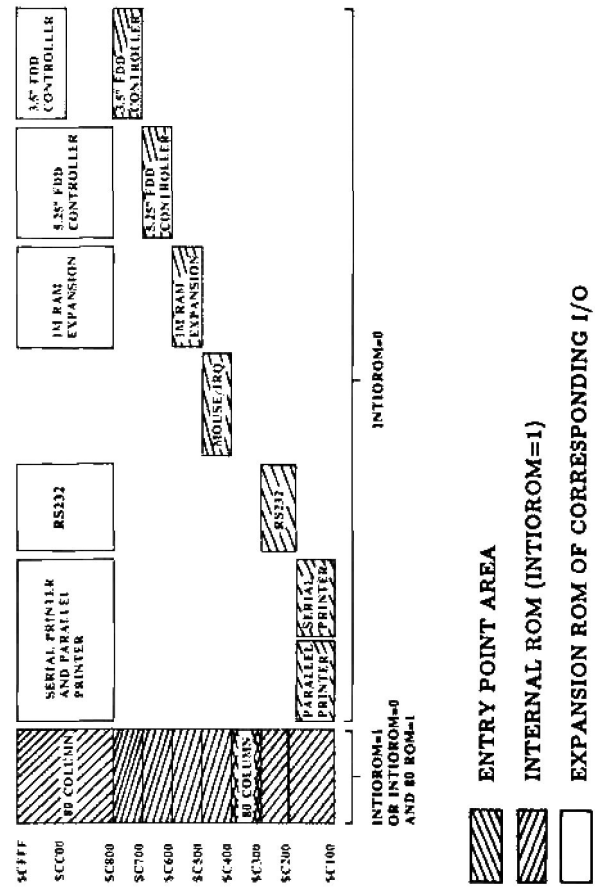


Fig. 3-3 I/O firmware areas

The only I/O firmware that remains active when INTIOROM is turned on is the 80-column display driver. When INTIOROM remains inactive, i.e. INTIOROM = 0, the possible I/O activities are:

- parallel printer
- serial printer
- RS232
- 80-column text
- mouse
- 1 M RAM expansion
- FDD (Floppy Disk Drive) controller (5.25"/3.5")

At any one time, only one kind of printer can be used (either parallel or serial). This is selected by a switch on the front panel of the computer.

3.2.3.2 I/O firmware control

a) INTIOROM = 1

If INTIOROM equals one, all the internal ROM area ($C100 to $CFFF) can be accessed freely.

b) INTIOROM = 0

If 80ROM is reset, then the internal $C3XX (for 80-column) and its expansion ROM cannot be accessed. Should 80ROM be active, the 80-column firmware can be accessed.

The 80-column expansion ROM can be accessed after reading from or writing to any location within $C300 - $C3FF. It is turned off by accessing $CFFF.

For all I/O firmwares, their expansion ROM can be turned off by accessing $CFFF.

If reading from or writing to that I/O firmware entry point region, their expansion ROM will be automatically turned on.

- The parallel printer and serial printer share the same expansion ROM ($C800 - $CFFF) so only one type of printer can be active at any one time.

- RS232 firmware occupies the memory address range $C200 - $C2FF.

- The mouse firmware only occupies $C400 - $C4FF. Accessing any location in this range cannot affect any $C800-$CFFF expansion ROM.

- The 1M RAM expansion firmware occupies locations from $C500 - $C5FF.

- The 5.25" floppy disk drive controller firmware occupies locations from $C600-$C6FF.

- The 3.5" disk drive firmware occupies locations from $C700 - $C7FF.

3.3 RAM

The RAM subsystem can be divided into:

a) Low bank main RAM ($0000 - $BFFF)
b) High bank main RAM ($D000 - $FFFF)
c) Low bank auxiliary RAM ($0000 - $BFFF)
d) High bank auxiliary RAM ($D000 - $FFFF)

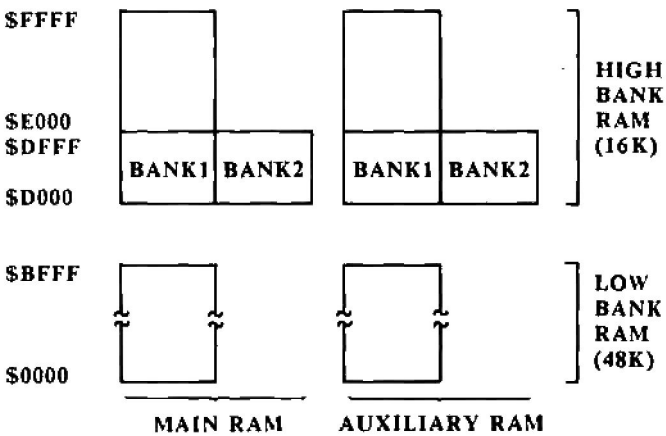


Fig. 3-4 RAM map

The operation of these memory banks will be discussed in detail in the following sections.

3.3.1 RAM memory usage

Some RAM memory is dedicated for special functions due to the CPU and system requirement. However most of the RAM memory are free areas where the user's program can use. By knowing more of the RAM functional partition, the user can manage the RAM more efficiently.

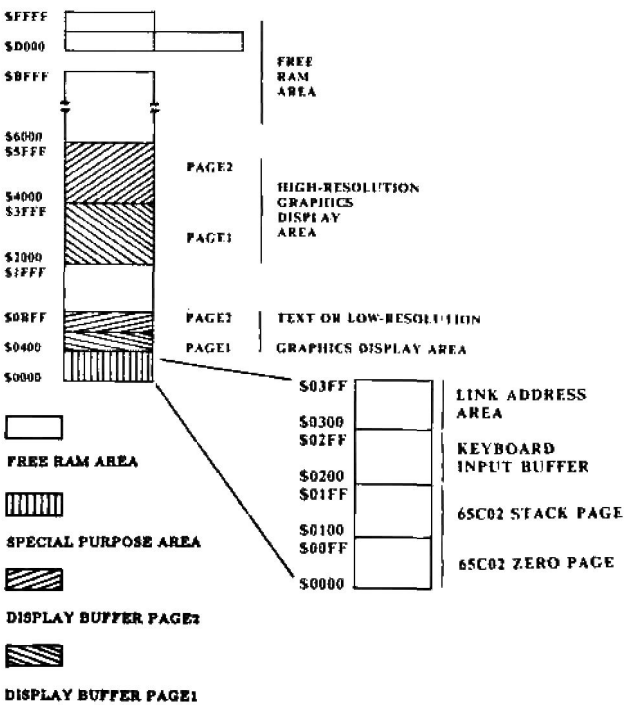


Fig. 3-5 RAM memory usage map

- Page one

  The 65C02 uses page 1 as its stack. When subroutine call or an interrupt occurs the microprocessor will push the return addresses onto the stack. Upon return from a subroutine or an interrupt the address will be pulled from stack. Many programs also use the stack for temporary storage of registers. The 65C02 operates the stack on a first-in, last-out basis.

- Page two

  This is the keyboard input buffer used by the Monitor and the BASIC interpreter.

- Page three

  The DOS and Monitor use it to store link addresses or vectors. Since only the upper part of page 3 is used by the computer, most of the page 3 area may be used freely.

ADDRESS	FUNCTION
03F0 03F1	BRK request vector used by Monitor
03F2 03F3	Reset vector
03F4	Power-up signature byte
03F5 03F6 03F7	Jump instruction to the subroutine that handles BASIC "&" commands
03F8 03F9 03FA	Jump instruction to the subroutine that handles user (CTRL - Y) commands
03FB 03FC 03FD	Jump instruction to the routine that handles non-maskable interrupt.
03FE 03FF	Interrupt Request vector

Table 3-1 Page three vectors used by Monitor

- Video buffers

  The text mode and low resolution graphics mode share the same display buffer. Page one is from $0400 - $07FF and page two is from $0800 - $0BFF. High resolution graphics page one uses $2000 - $3FFF and page two is from $4000 - $5FFF. If the display buffer is not used for display, it can be used by the user as program area.

- Free RAM area ($6000 - $BFFF)

  The user's program can be freely placed in this RAM area.

3.3.2 High-bank RAM memory ($D000 - $FFFF)

This RAM memory refers to address $D000 - $FFFF. An additional 4K RAM is placed next to $D000 - $DFFF as BANK 2.

This high-bank RAM can be freely selected as read-only, write-only, read/write or inhibited totally. It should be noted that this high-bank RAM occupies the same address range as ROM. Moreover, there are separate high-bank RAM areas for main and auxiliary RAM memory.

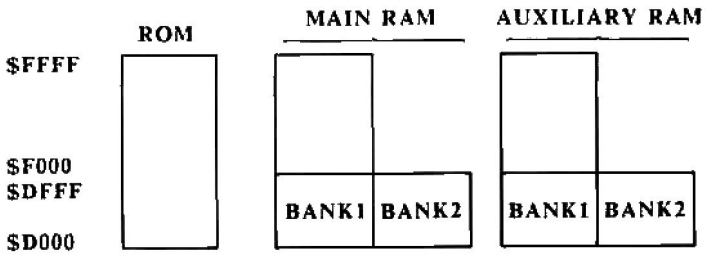


Fig. 3-6 High bank RAM memory mapping

3.3.2.1 High - Bank RAM Switches

To control the operation of the high-bank RAM, the switches in table 3-2 can be used. When power is turned on or CTRL-RESET is pressed, the high-bank RAM will always be set to write-enabled and ROM set to read enabled.

Operation	Address	Function
R	$C080 (or $C084)	Read high-bank RAM only; bank2
R TWICE	$C081 (or $C085)	Read ROM, write high-bank RAM; bank2

Operation	Address	Function
R	$C082 or $C086)	Read ROM only
R twice	$C083 (or $C087)	Read and write high-bank RAM; bank2
R	$C088 (or $C08C)	Read high-bank RAM only; bank1
R TWICE	$C089 (or $C08D)	Read ROM, write high-bank RAM; bank1
R	$C08A (or $C08E)	Read ROM only
R TWICE	$C08B (or $C08F)	Read and write high-bank RAM; bank1
W	$C008 (AUXZP)	Off AUXZP; use main high-bank, page 0 & page 1

Operation	Address	Function
W	$C009 (AUXZP)	On AUXZP; use auxiliary high-bank, page 0 & page 1
R7	$C016 (Read AUXZP)	If bit 7= 1 then auxiliary bank RAM is in use If bit 7=0 then main bank RAM is in use.

Table 3-2 RAM control hardware locations for Zero Page and high-bank (N.B. R=Read; W=write ; R7=read bit 7 of the location)

How the switches affect the high-bank memory are illustrated in Fig 3-7 to Fig 3-10.

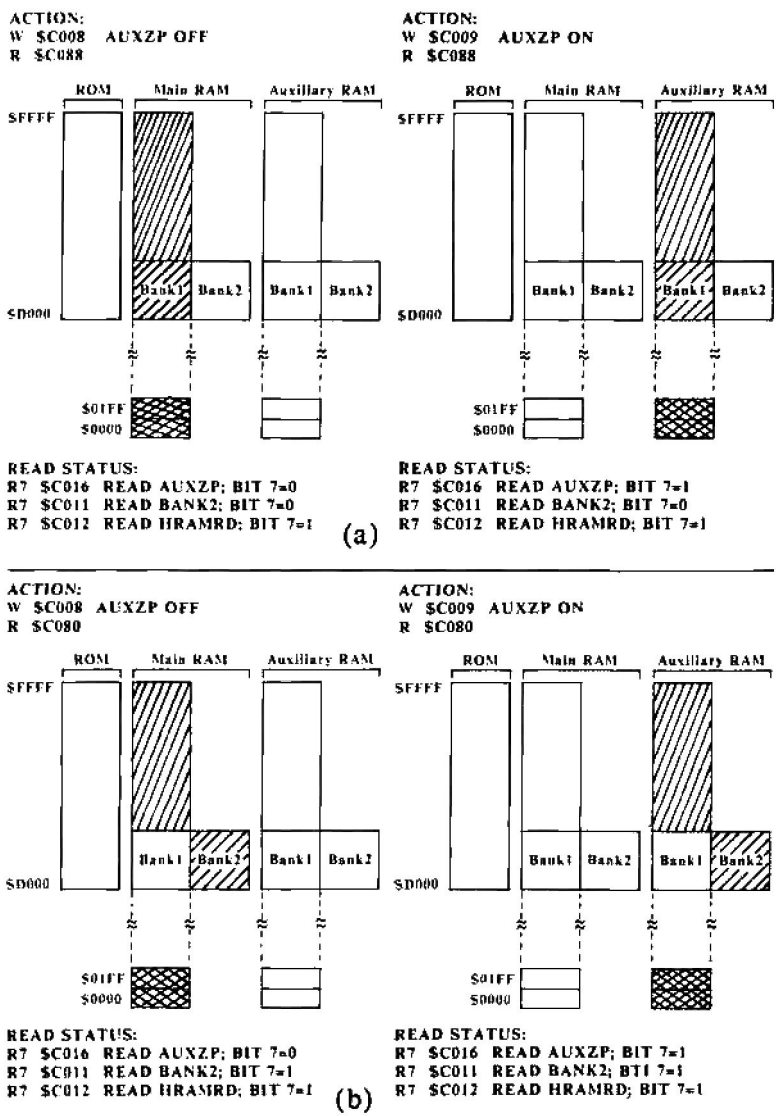


Fig. 3-7 High bank and zero page memory.

(a) Read high-bank RAM bank 1 memory
(b) Read high-bank RAM bank 2 memory

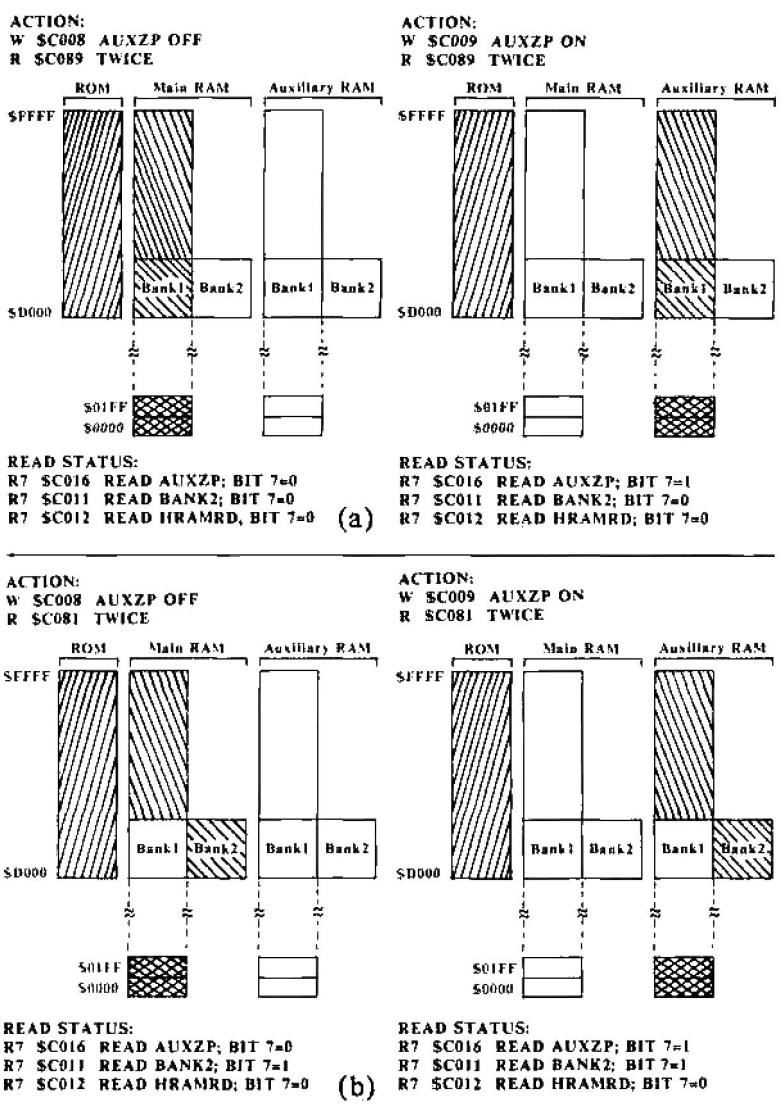


Fig. 3-8 High-bank and zero page memory.

(a) Read ROM; write high-bank RAM bank 1 memory

(b) Read ROM; write high-bank RAM bank 2 memory

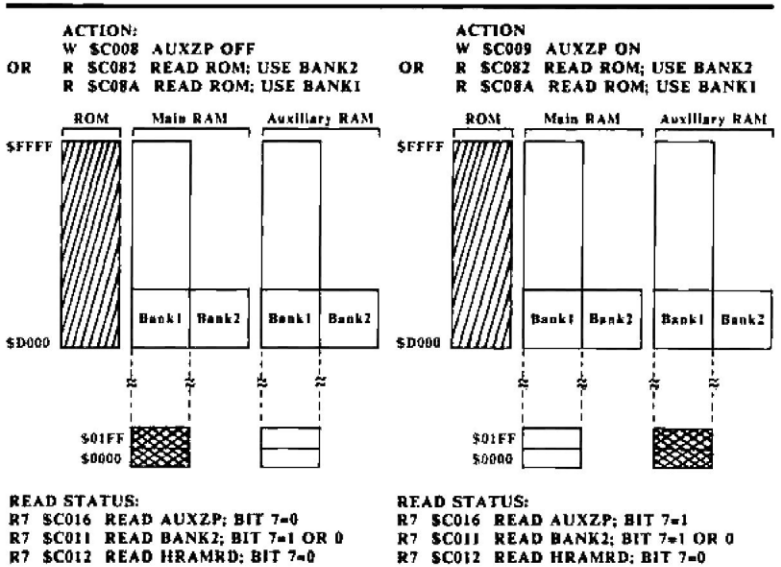
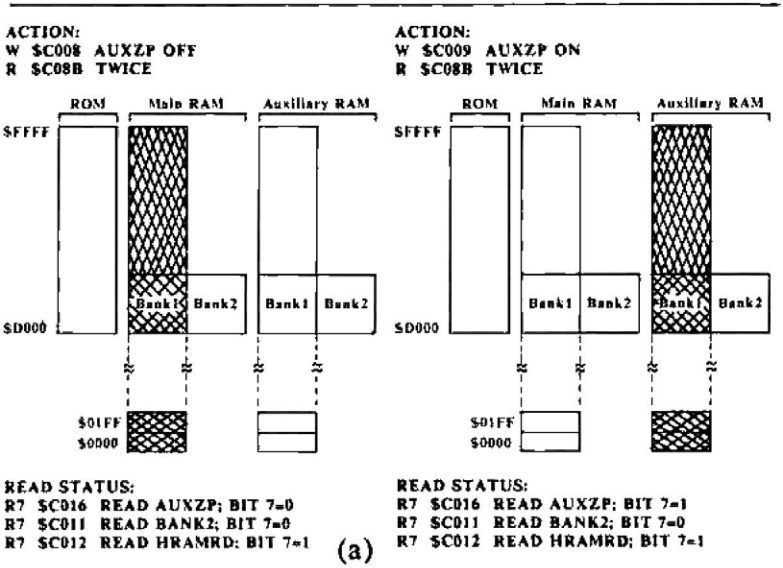


Fig. 3-9 High bank and zero page memory (Read ROM only)



(a)

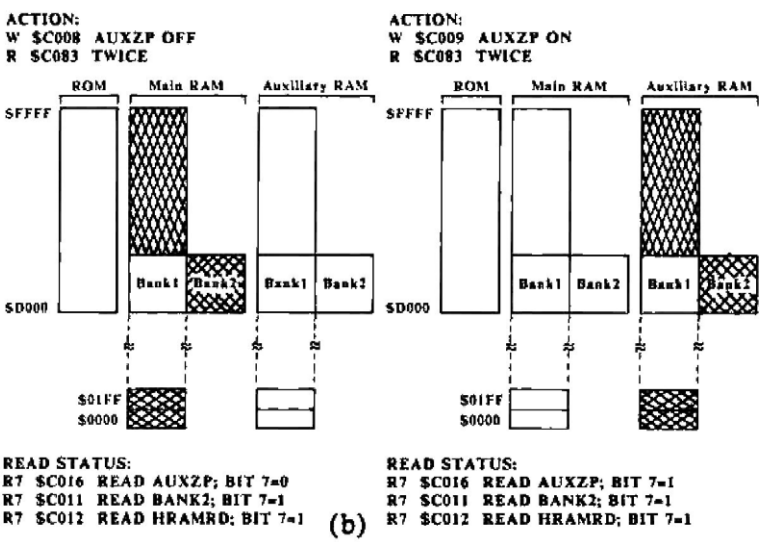


Fig. 3-10 High bank memory:

(a) Read and write high-bank RAM bank1 memory
(b) Read and write high-bank RAM bank2 memory

Read

Write

3.3.3 Low-bank RAM memory ($0000 - $BFFF)

The low-bank RAM can be divided into 2 parts according to bank control switches:

- Zero page ($0000 - $00FF) RAM and the stack ($0100 - $01FF)
- RAM addresses from $0200 - $BFFF

3.3.3.1 Zero page RAM and the stack

This part of RAM can be controlled by the AUXZP switch as the high-bank RAM range. When AUXZP soft-switch is turned on, the auxiliary RAM page zero and stack ($0000 - $01FF) is read and write accessible. If AUXZP is off, the main RAM ($0000 - $01FF) is read and write accessible. The function of the soft-switches can be found in Fig. 3-7 to 3-10.

The user is advised to take care in using zero page because the 65C02 and system software use this area for storage of important parameters.

3.3.3.2 $0200 - $BFFF RAM memory

The switches which affect the read and write operation on this memory bank are shown in Table 3-3. However, the display buffer memory of either graphics or text mode is also within the range of $0200 to $BFFF. The display buffer memory switches have higher priority over memory control switches in Table 3-3, i.e. when the display buffer memory switches are effective, the ARAMRD and ARAMWR switches lose control over the display buffer memory. The display buffer memory will be discussed fully in the next section.

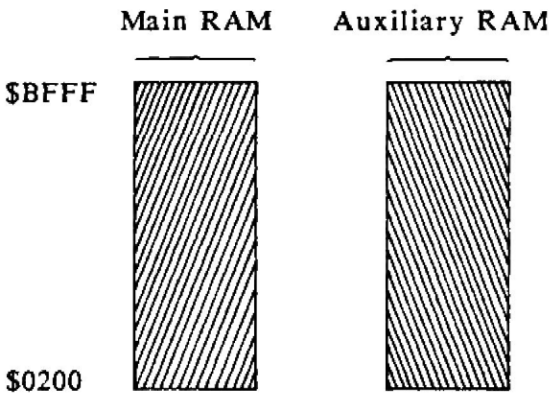
Address	Operation	Function
$C002	W	off ARAMRD; read main RAM ($0200-$BFFF)
$C003	W	on ARAMRD; read auxiliary RAM ($0200-$BFFF)
$C004	W	off ARAMWR; write main RAM ($0200-$BFFF)

Address	Operation	Function
$C005	W	on ARAMWR; write auxiliary RAM ($0200 - $BFFF)
$C013	R7	read ARAMRD status: If bit 7=1; read auxiliary RAM. If bit 7=0; read main RAM
$C014	R7	read ARAMWR status: If bit 7=1; write auxiliary RAM If bit 7=0; write main RAM.

Table 3-3 $0200-$BFFF RAM bank switches

The ARAMRD and ARAMWR switches affect the READ / WRITE status of the $0200-$BFFF bank of RAM. These two switches operate independently. E.g. we can set main-RAM-read and auxiliary-RAM-write.

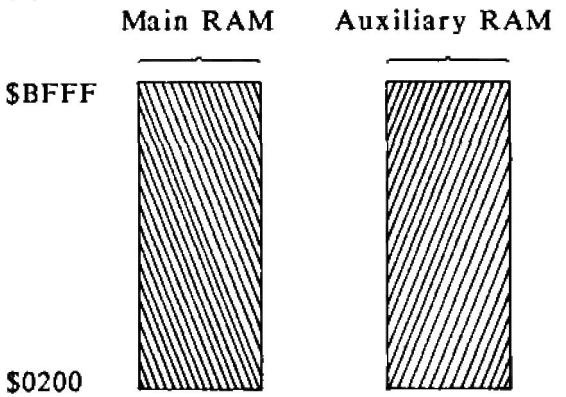
(a)



Operation:
W $C000 ; off INHPAGE2
W $C002 ; read main RAM
W $C005 ; write auxiliary RAM

Read Status:
R7 $C018 ; read INHPAGE2 bit 7=0
R7 $C013 ; read ARAMRD bit 7=0
R7 $C014 ; read ARAMWR bit 7=1

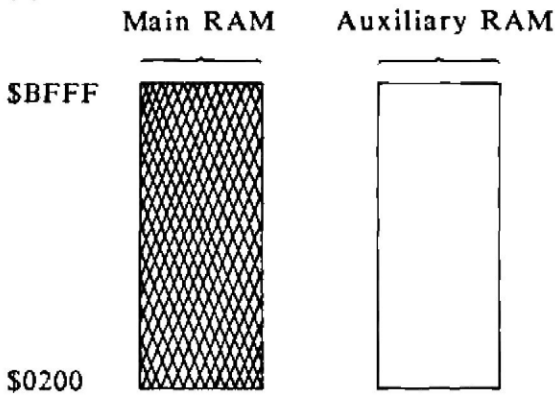
(b)



Operation:
W $C000 ; INHPAGE2 off
W $C003 ; read auxiliary RAM
W $C004 ; write main RAM

Read Status:
R7 $C018 ; read INHPAGE2 bit 7=0
R7 $C013 ; read ARAMRD bit 7=1
R7 $C014 ; read ARAMWR bit 7=0

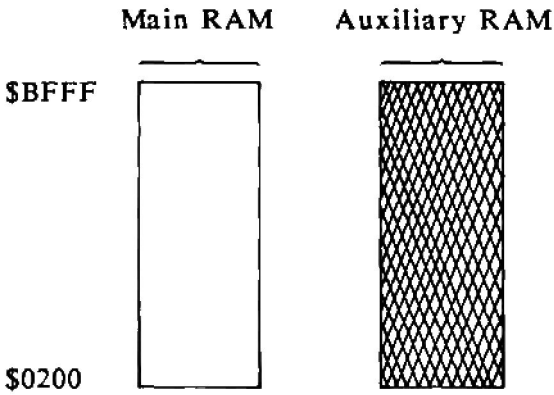
(c)



Operation:
W $C000 ; INHPAGE2 off
W $C002 ; read main RAM
W $C004 ; write main RAM

Read Status:
R7 $C018 ; read INHPAGE2 bit 7=0
R7 $C013 ; read ARAMRD bit 7=0
R7 $C014 ; read ARAMWR bit 7=0

(d)



Operation:
W $C000 ; INHPAGE2 off
W $C003 ; read auxiliary RAM
W $C005 ; write auxiliary RAM

Read Status:
R7 $C018 ; read DOUBLE bit 7=0
R7 $C013 ; read ARAMRD bit 7=1
R7 $C014 ; read ARAMWR bit 7=1

Fig 3-11 $0200-$BFFF Bank RAM

a) Read main RAM, write auxiliary RAM
b) Read auxiliary RAM, write main RAM
c) Read and write main RAM
d) Read and write auxiliary RAM

3.3.3.3 Display buffer memory

Fig. 3-12 illustrates the memory range covered by display buffer memory.

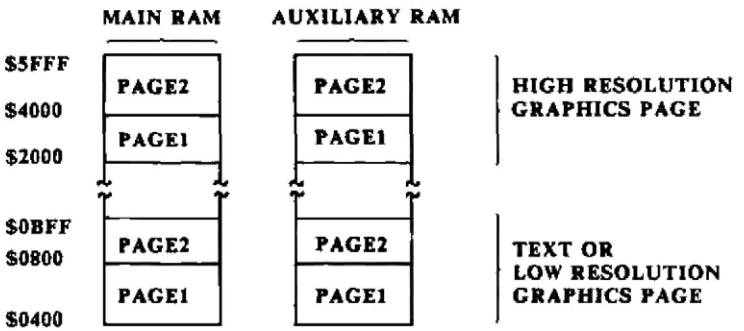


Fig. 3-12 Display buffer memory mapping

Since the display buffer memory overlaps with part of the low bank RAM, when the display buffer memory soft-switches are active, ARAMRD and ARAMWR will have no effect on these memory areas.

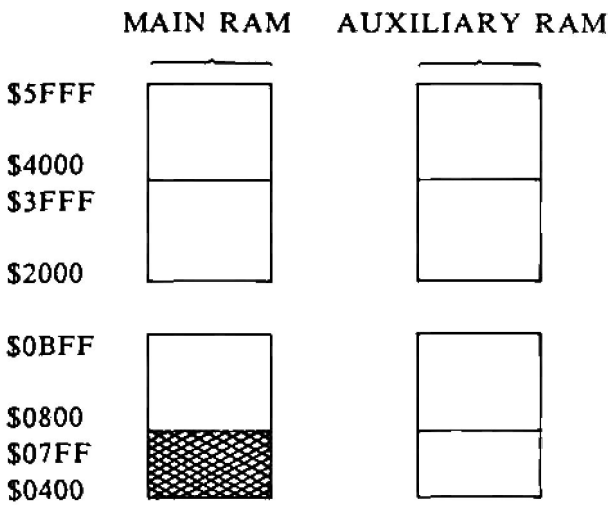
The switches affecting the display buffer memory are shown in Table 3-4.

Address	Operation	Function
$C000	W	off INHPAGE2; ARAMRD and ARAMWR control the display memory
$C001	W	on INHPAGE2; DPAGE2 and HGR control the display memory
$C054	R/W	off DPAGE2; refer to Fig. 3-13
$C055	R/W	on DPAGE2; refer to Fig. 3-13
$C056	R/W	off HGR: turns off high resolution graphics; refer to Fig. 3-13

Address	Operation	Function
$C057	R/W	on HGR; turns on high resolution graphics refer to Fig. 3-13
$C018	R7	Read INHPAGE2 status: If bit 7 = 1; INHPAGE2 on If bit 7 = 0; INHPAGE2 off
$C01C	R7	Read DPAGE2 status: If bit 7 = 1; DPAGE2 on If bit 7 = 0; DPAGE2 off
$C01D	R7	Read HGR status: If bit 7 = 1; HGR on If bit 7 = 0; HGR off

Table 3-4 Display buffer memory switches

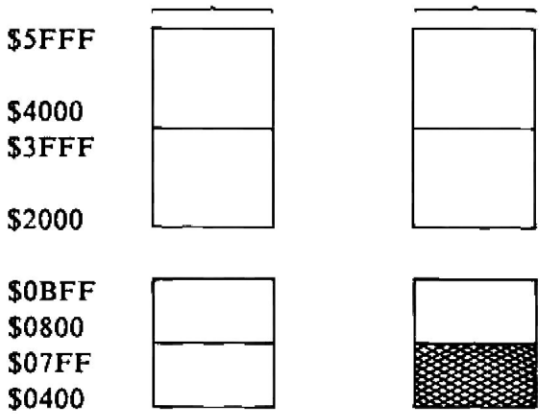
In Fig. 3-13 the operational relation between INHPAGE2, DPAGE2 and HGR is shown. It should be noted that when INHPAGE2 is off, ARAMRD and ARAMWR control the reading or writing of the RAM. The DPAGE2 switch only affects the display. The relation between DPAGE2 and display memory is shown in Fig. 3-14.



Operation:
W $C001; on INHPAGE 2
W $C054; off DPAGE2
W $C056; off HGR

Read status:
R7 $C018; Read INHPAGE2; bit 7=1
R7 $C01C; Read DPAGE2; bit 7=0
R7 $C01D; Read HGR; bit 7=0

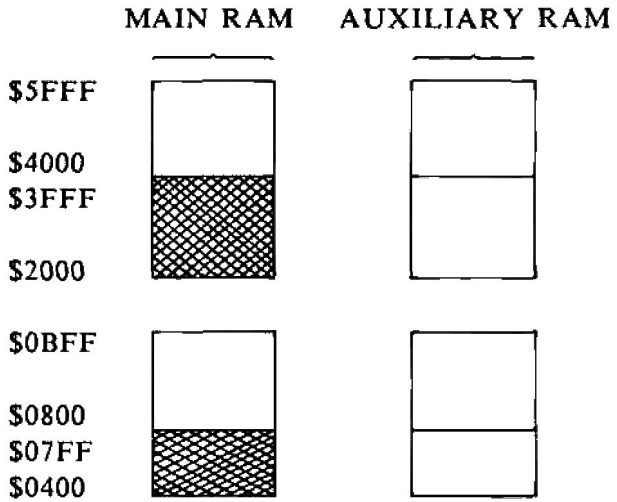
(a) INHPAGE2 on HGR off and DPAGE2 off



Operation:
W $C001; on INHPAGE2
W $C055; on DPAGE2
W $C056; off HGR

Read status:
R7 $C018; Read INHPAGE2; bit 7=1
R7 $C01C; Read DPAGE2; bit 7=1
R7 $C01D; Read HGR; bit 7=0

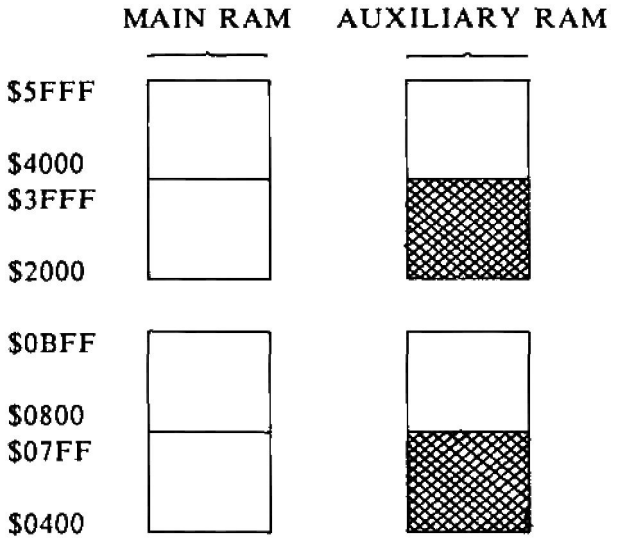
(b) INHPAGE2 on, HGR off and DPAGE2 on



Operation:
W $C001; on INHPAGE2
W $C054; off DPAGE2
W $C057; on HGR

Read status:
R7 $C018; Read INHPAGE2; bit 7=1
R7 $C01C; Read DPAGE2; bit 7=0
R7 $C01D; Read HGR; bit 7=1

(c) INHPAGE2 on, HGR on and DPAGE2 off.



Operation:
W $C001; on INHPAGE2
W $C055; on DPAGE2
W $C057; on HGR

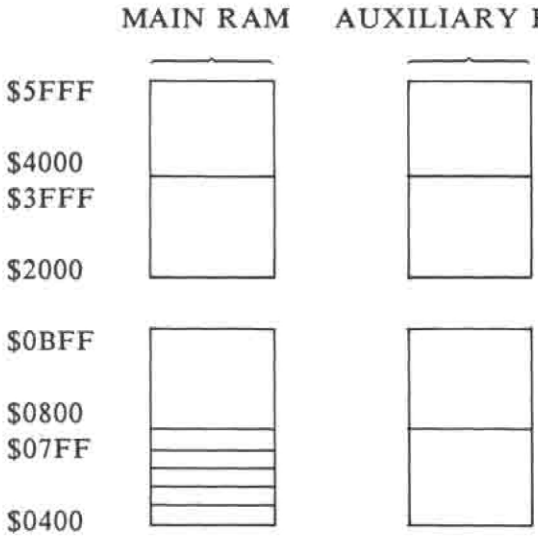
Read status:
R7 $C018; Read INHPAGE2; bit 7=1
R7 $C01C; Read DPAGE2; bit 7=1
R7 $C01D; Read HGR; bit 7=1

RAM read and write

(d) INHPAGE2 on, HGR on and DPAGE2 on.

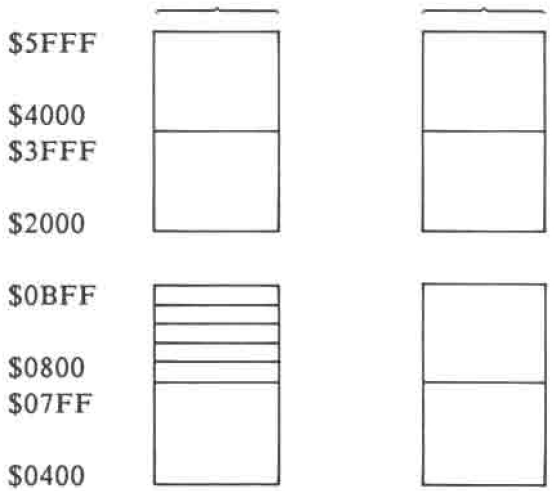
Fig. 3-13 Display buffer switching

* When INHPAGE2 is off, ARAMRD and ARAMWR will control which bank of RAM to be read or written.



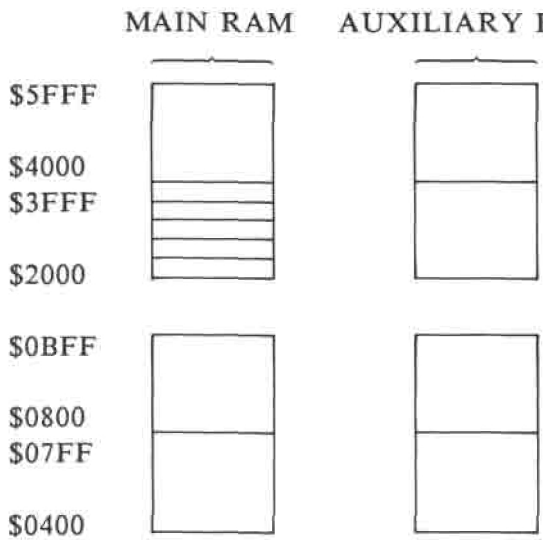
Operation:
R/W $C054; off DPAGE2
R/W $C056; off HGR

(a) INHPAGE2 off, HGR off and DPAGE2 off.



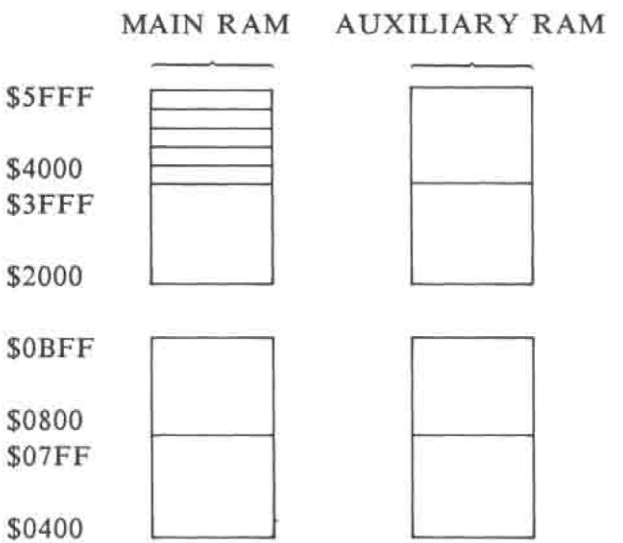
Operation:
R/W $C055; on DPAGE2
R/W $C056; off HGR

(b) INHPAGE2 off, HGR off and DPAGE2 on.



Operation:
R/W $C054; off DPAGE2
R/W $C057; on HGR
R/W $C050; off TEXT

(c) INHPAGE2 off, HGR on and DPAGE2 off.



Operation:
R/W $C055; on DPAGE2
R/W $C057; on HGR
R/W $C050; off TEXT

memory to be displayed on screen

Fig. 3-14 Selection of video display buffer

When in double resolution modes, display buffer ($0400-$07FF or $2000-$3FFF) of auxiliary RAM will display simultaneously with display buffer ($0400-$07FF or $2000-$3FFF) of main RAM. The setting of DPAGE2 will lose control in these modes.

3.4 Hardware page ($C000 - C0FF)

There is no physical RAM or ROM in this page. This page is important to the operation of the system because all the soft-switches and their status bits are in this page. The operation of these soft-switches can be found in details in various chapters of this manual.

Location	Operation	Description
C00X	R	BIT 7=KEY STROBE; BIT 0-6 KEYBOARD DATA
C000	W	OFF INHPAGE2
C001	W	ON INHPAGE2
C002	W	OFF ARAMRD
C003	W	ON ARAMRD
C004	W	OFF ARAMWR
C005	W	ON ARAMWR
C006	W	OFF INTIOROM
C007	W	ON INTIOROM
C008	W	OFF AUXZP

Location	Operation	Description
C009	W	ON AUXZP
C00A	W	OFF 80ROM
C00B	W	ON 80ROM
C00C	W	OFF DBLRES
C00D	W	ON DBLRES
C00E	W	OFF CHARSET2
C00F	W	ON CHARSET2
C01X	W	RESET KEY STROBE
C010	R	BIT 7=1; A KEY IS BEING PRESSED
C011	R	BIT 7=1; BANK2 ON
C012	R	BIT 7=1; HRAMRD ON
C013	R	BIT 7=1; ARAMRD ON
C014	R	BIT 7=1; ARAMWR ON

Location	Operation	Description
C015	R	BIT 7=1; INTIOROM ON
C016	R	BIT 7=1; AUXZP ON
C017	R	BIT 7=1; 80ROM ON
C018	R	BIT 7=1; INHPAGE2 ON
C019	R	BIT 7=1; VERTICAL BLANKING NOT ACTIVE
C01A	R	BIT 7=1; TEXT ON
C01B	R	BIT 7=1; MIX ON
C01C	R	BIT 7=1; DPAGE2 ON
C01D	R	BIT 7=1; HGR ON
C01E	R	BIT 7=1; CHARSET2 ON

Location	Operation	Description
C01F	R	BIT 7=1; DBLRES ON
C02X	R/W	RESERVED
C03X	R/W	TOGGLE SPEAKER OUTPUT
C04X	R/W	RESERVED
C050	R/W	OFF TEXT
C051	R/W	ON TEXT
C052	R/W	OFF MIX
C053	R/W	ON MIX
C054	R/W	OFF DPAGE2
C055	R/W	ON DPAGE2
C056	R/W	OFF HGR
C057	R/W	ON HGR
C058-C05D	R/W	RESERVED
C05E	R/W	OFF INHDRGR

Location	Operation	Description
C05F	R/W	ON INHDRGR
C06X	W	RESERVED
C060	R	BIT 7=1; 40/80 SWITCH TO 40 POSITION
C061	R	READ SWITCH INPUT 0 OR △
C062	R	READ SWITCH INPUT 1 OR ▲
C063	R	READ MOUSE BUTTON
C064	R	READ TIMER 0
C065	R	READ TIMER 1
C066	R	READ MOUSE XDIR
C067	R	READ MOUSE YDIR
C068-C06F	R	RESERVED

Location	Operation	Description
C07X	R/W	RESET VERTICAL BLANKING INTERRUPT AND JOYSTICK PORT TIMERS
C080-C08F	W	RESERVED
C080	R	READ HIGH BANK 2 RAM
C081	R TWICE	READ ROM AND WRITE HIGH BANK 2 RAM
C082	R	READ ROM
C083	R TWICE	READ AND WRITE HIGH BANK 2 RAM
C084-C087	R	REPEAT C080-C083 FUNCTION
C088	R	READ HIGH BANK 1 RAM

Location	Operation	Description
C089	R TWICE	READ ROM AND WRITE HIGH BANK 1 RAM
C08A	R	READ ROM
C08B	R TWICE	READ AND WRITE HIGH BANK 1 RAM
C08C-C08F	R	REPEAT C088-C08B FUNCTION
C090-C097	R/W	RESERVED
C098	R/W	ACIA1 RECEIVE/ TRANSMIT DATA REGISTER
C099	R/W	ACIA1 STATUS REGISTER
C09A	R/W	ACIA1 COMMAND REGISTER

Location	Operation	Description
C09B	R/W	ACIA1 CONTROL REGISTER
C09C-C09F	R/W	RESERVED
C0A0-C0A7	R/W	RESERVED
C0A8	R/W	ACIA2 RECEIVE/ TRANSMIT DATA REGISTER
C0A9	R/W	ACIA2 STATUS REGISTER
C0AA	R/W	ACIA2 COMMAND REGISTER
C0AB	R/W	ACIA2 CONTROL REGISTER
C0AC-C0AF	R/W	RESERVED
C0BX	R/W	RESERVED

Location	Operation	Description
C0C0-C0C7	R	RESERVED
C0C0	W	ON MOUSE X-DIR RISING-EDGE INTERRUPT
C0C1	W	ON MOUSE X-DIR FALLING-EDGE INTERRUPT
C0C2	W	ON MOUSE Y-DIR RISING-EDGE INTERRUPT
C0C3	W	ON MOUSE Y-DIR FALLING-EDGE INTERRUPT
C0C4	W	OFF MOUSE INTERRUPT SOURCE
C0C5	W	ON MOUSE INTERRUPT SOURCE

Location	Operation	Description
C0C6	W	OFF VERTICAL BLANKING INTERRUPT
C0C7	W	ON VERTICAL BLANKING INTERRUPT
C0C8-C0CE	W	RESERVED
C0C8	R	BIT 7=1/0; MOUSE X-DIR FALLING/ RISING EDGE INTERRUPT SELECTED
C0C9	R	BIT 7=1/0; MOUSE Y-DIR FALLING/ RISING EDGE INTERRUPT SELECTED
C0CA	R	BIT 7=1; MOUSE INTERRUPT ENABLED

Location	Operation	Description
C0CB	R	BIT 7=1; VERTICAL BLANKING INTERRUPT ENABLED
C0CC	R	BIT 7=1; MOUSE X-DIR INTERRUPT OCCURRED
C0CD	R	BIT 7=1; MOUSE Y-DIR INTERRUPT OCCURRED
C0CF	R	RESERVED
C0CF	W	RESET MOUSE INTERRUPT
C0D0-C0D3	R/W	RESERVED FOR EXPANSION RAM CONTROL
C0D4-C0DF	R/W	RESERVED

Location	Operation	Description
C0EX	R/W	RESERVED FOR FLOPPY DISK DRIVE CONTROL
C0FX	R/W	RESERVED

NOTE 1: Read $C010 to $C01F will read the keyboard code (bit 0-bit 6) and reading $C010 resets KEYSTROBE too.

NOTE 2: $C081, $C083, $C089 and $C08B have to be read twice to achieve the described function in the table.

Table 3-5 Hardware page locations.

CHAPTER 4
KEYBOARD AND SPEAKER

4. KEYBOARD AND SPEAKER

Among the various I/O devices for the computer, the keyboard, speaker and video display are most commonly used. In this chapter, we will focus on the keyboard and the speaker, while the video display will be discussed in the next chapter.

4.1 Keyboard

The keyboard of the computer has a typewriter layout, a numeric keypad and ten function keys. The specifications of the keyboard are listed in Table 4-1.

Number of keys	:	90
Encoding format of character	:	ASCII
Special keys	:	10 function keys, ▲ and △ CONTROL - RESET
Cursor keys	:	← → ↑ ↓
Features	:	Auto-repeat

Table 4-1 Keyboard specifications

The computer keyboard layout can be changed by the keyboard switch on the back panel. Keyboard layout is selectable between standard U.S.A. and alternate Dvorak layout using the STD/ALT KBD switch on the back panel of the computer.
(NOTE: some versions do not have the keyboard switch and only standard U.S.A. layout is available).

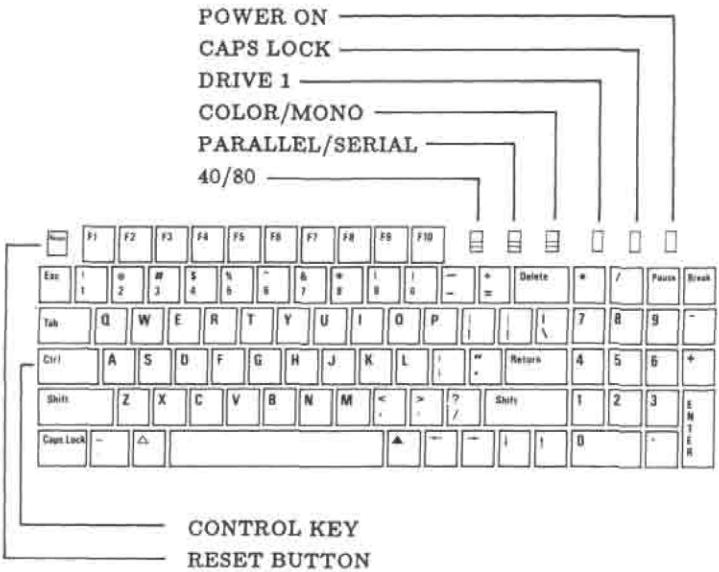


Fig. 4-1 Keyboard switches and indicator lights.

Fig. 4-2 (a) Keyboard switch set to STD (The USA standard keyboard)

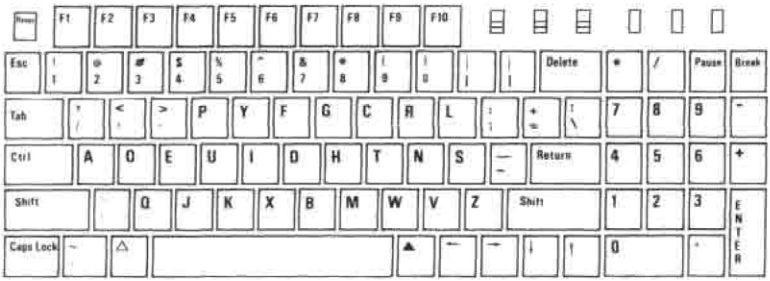


Fig. 4-2 (b) Keyboard switch is set to ALT (The simplified keyboard)

There are 3 indicator lights on the front panel. They are

a) POWER - when power is turned on, the light will be on

b) DISK - when the built-in drive is accessed, this light will glow.

c) CAPSLOCK - when this light glows, it indicates that the keyboard is in capslock mode, that is all letter keys will produce capital letters on the screen irrespective of the position of the shift keys.

The keyboard will generate ASCII code when any one key of the keyboard is pressed. The keys to generate ASCII codes are tabulated in Table 4-2.

Table 4-2 Keys & the ASCII codes

KEY	NORMAL	CHAR	CTRL	CHAR	SHIFT	CHAR	BOTH	CHAR
DELETE	7F	DEL	7F	DEL	7F	DEL	7F	DEL
←	08	BS	08	BS	08	BS	08	BS
TAB	09	HT	09	HT	09	HT	09	HT
↓	0A	LF	0A	LF	0A	LF	0A	LF
↑	0B	VT	0B	VT	OB	VT	0B	VT
RETURN	0D	CR	0D	CR	0D	CR	0D	CR
→	15	NAK	15	NAK	15	NAK	15	NAK
ESC	1B	ESC	1B	ESC	1B	ESC	1B	ESC
SPACE	20	SP	20	SP	20	SP	20	SP
' "	27	'	27	'	22	"	22	"
, <	2C	,	2C	,	3C	<	3C	<
- _	2D	-	1F	US	5F	_	1F	US
. >	2E	.	2E	.	3E	>	3E	>
/ ?	2F	/	2F	/	3F	?	3F	?
0)	30	0	30	0	29	)	29	)
1 !	31	1	31	1	21	!	21	!
2 @	32	2	00	NUL	40	@	00	NUL
3 #	33	3	33	3	23	#	23	#
4 $	34	4	34	4	24	$	24	$
5 %	35	5	35	5	25	%	25	%
6 ^	36	6	1E	RS	5E	^	1E	RS
7 &	37	7	37	7	26	&	26	&
8 *	38	8	38	8	2A	*	2A	*
9 (	39	9	39	9	28	(	28	(
; :	3B	;	3B	;	3A	:	3A	:
= +	3D	=	3D	=	2B	+	2B	+
[{	5B	[	1B	ESC	7B	{	1B	ESC
\ |	5C	\	1C	FS	7C	|	1C	FS
] }	5D	]	1D	GS	7D	}	1D	GS
` ~	60	`	60	`	7E	~	7E	~
A	61	a	01	SOH	41	A	01	SOH
B	62	b	02	STX	42	B	02	STX
C	63	c	03	ETX	43	C	03	ETX
D	64	d	04	EOT	44	D	04	EOT
E	65	e	05	ENQ	45	E	05	ENQ
F	66	f	06	ACK	46	F	06	ACK
G	67	g	07	BEL	47	G	07	BEL
H	68	h	08	BS	48	H	08	BS
I	69	i	09	HT	49	I	09	HT
J	6A	j	0A	LF	4A	J	0A	LF
K	6B	k	0B	VT	4B	K	0B	VT

KEY	NORMAL	CHAR	CTRL	CHAR	SHIFT	CHAR	BOTH	CHAR
L	6C	l	0C	FF	4C	L	0C	FF
M	6D	m	0D	CR	4D	M	0D	CR
N	6E	n	0E	SO	4E	N	0E	SO
O	6F	o	0F	SI	4F	O	0F	SI
P	70	p	10	DLE	50	P	10	DLE
Q	71	q	11	DC1	51	Q	11	DC1
R	72	r	12	DC2	52	R	12	DC2
S	73	s	13	DC3	53	S	13	DC3
T	74	t	14	DC4	54	T	14	DC4
U	75	u	15	NAK	55	U	15	NAK
V	76	v	16	SYN	56	V	16	SYN
W	77	w	17	ETB	57	W	17	ETB
X	78	x	18	CAN	58	X	18	CAN
Y	79	y	19	EM	59	Y	19	EM
Z	7A	z	1A	SUB	5A	Z	1A	SUB
0	30	0	30	0	30	0	30	0
1	31	1	31	1	31	1	31	1
2	32	2	32	2	32	2	32	2
3	33	3	33	3	33	3	33	3
4	34	4	34	4	34	4	34	4
5	35	5	35	5	35	5	35	5
6	36	6	36	6	36	6	36	6
7	37	7	37	7	37	7	37	7
8	38	8	38	8	38	8	38	8
9	39	9	39	9	39	9	39	9
.	2E	.	2E	.	2E	.	2E	.
+	2B	+	2B	+	2B	+	2B	+
-	2D	-	2D	-	2D	-	2D	-
*	2A	*	2A	*	2A	*	2A	*
/	2F	/	2F	/	2F	/	2F	/
PAUSE	13	DC3	13	DC3	13	DC3	13	DC3
BREAK	03	ETX	03	ETX	03	ETX	03	ETX
ENTER	0D	CR	0D	CR	0D	CR	0D	CR
F1	00	NUL	0	NUL	0	NUL	0	NUL
F2	01	SOH	1	SOH	1	SOH	1	SOH
F3	02	STX	2	STX	2	STX	2	STX
F4	03	ETX	3	ETX	3	ETX	3	ETX
F5	04	EOT	4	EOT	4	EOT	4	EOT
F6	05	ENQ	5	ENQ	5	ENQ	5	ENQ
F7	06	ACK	6	ACK	6	ACK	6	ACK
F8	07	BEL	7	BEL	7	BEL	7	BEL
F9	0C	FF	0C	FF	0C	FF	0C	FF
F10	18	CAN	18	CAN	18	CAN	18	CAN

4.1.1 Accessing the keyboard

The keyboard input status and the ASCII keycode can be accessed by reading the hardware locations in Table 4-3.

Address	Operation	Function
$C00X	R7	Bit 7=1; a key has been pressed Bit 7=0; no key has ever been pressed Bit 0-6 ASCII code of the pressed key
$C010	R7	Bit 7=1; Any key is being pressed down Bit 7=0; No key is being pressed down
$C01X	R	Bit 0-6; ASCII code of the pressing key

Address	Operation	Function
$C010	R/W	reset the keyboard strobe latch
$C01X	W	reset the keyboard strobe latch.

Table 4-3 keyboard hardware locations

When any key is pressed, the keyboard strobe latch is set to 1. This keyboard strobe status can be read at bit 7 of location $C00X. At the same time, the ASCII code of that key is also contained in bit 0-6. Reading $C01X can also read the ASCII code, but it will reset the keyboard strobe status. To check if any key is being pressed down, the location $C010 can be read but it will reset the keyboard strobe status at the same time. Usually we can check $C00X first to see if any key has been pressed before, then we can read the ASCII code by reading $C01X. Fig. 4-3 illustrates this event.

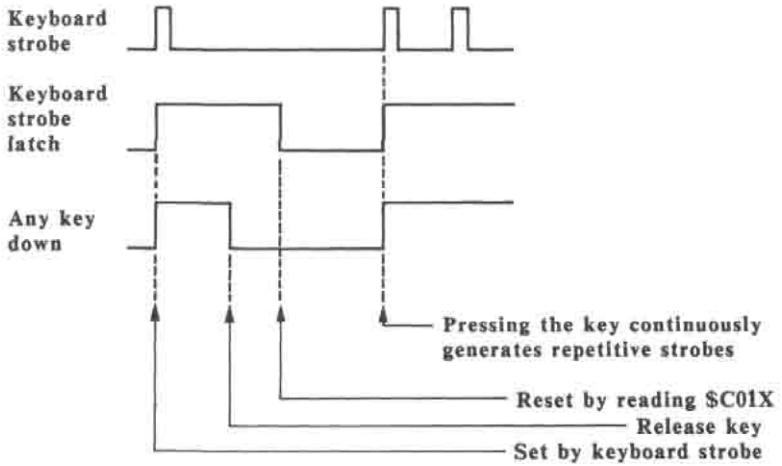


Fig. 4-3 Timing of pressing a key on the keyboard

The keyboard encoding IC has an auto-repeat function. If a key is constantly held down, then that key will be generated automatically at a fixed rate.

4.1.2 Special function keys and switches

There are some keys which do not generate any ASCII code. These keys affects the system immediately.

a) [CTRL] - [RESET]

Pressing these two keys simultaneously will reset the microcomputer system.

b) [△] , [▲]

The status of these two keys can be read from locations $C061 and $C062 respectively. These two keys can be used as game control keys.

c) [CAPS LOCK]

This is a toggling switch. When upper-case letters are activated, the CAPS LOCK indication light will glow.

d) 40 / 80 switch

80 column text mode can be turned on only when the 40 / 80 switch is set to 80 position.

Address	Operation	Function
$C060	R7	BIT 7=1; 40/80 switch set to 40 position BIT 7=0; 40/80 switch set to 80 position
$C061	R7	BIT 7=1;△pressed
$C062	R7	BIT 7=1;▲pressed

Table 4-4 Special keys and switch location

4.2 Speaker

Address	Operation	Function
$C03X	R	Speaker toggles when this is accessed

Table 4-5 Sound control hardware location

A speaker is built-in. Sound of different tones will be generated depending on the rate $C03X is being accessed. The loudness of the speaker can be adjusted by the volume control. The user can also use the ear-phone jack if he doesn't want to use the internal speaker.

CHAPTER 5
THE VIDEO DISPLAY

5. THE VIDEO DISPLAY

Besides conventional text display, the computer is also capable of displaying high-resolution color graphics. The following is a brief summary of the features of the video display:

- 40-column x 24-row text
- 80-column x 24-row text
- 40H x 24V, 16-color low-resolution graphics
- 80H x 24V, 16-color double-low-resolution graphics
- 280H x 192V, 6-color high-resolution graphics
- 560H x 192V, 16-color double-high-resolution graphics
- Mixed text/graphics display
- Secondary display page

The video display is controlled by a number of "software-switches" resided in the hardware page described in chapter 3. Table 5-1 lists the locations and functions of the various video display control switches while Table 5-2 and 5-3 shows the switch settings for selecting the various video display modes and display pages.

Address	Operation	Switch	State	Function
$C000	W	INHPAGE2	off	Enable display page 2
$C001	W	INHPAGE2	on	Inhibit display page 2
$C00C	W	DBLRES	off	Select normal resolution
$C00D	W	DBLRES	on	Select double resolution
$C00E	W	CHARSET2	off	Select character set 1
$C00F	W	CHARSET2	on	Select character set 2
$C050	R/W	TEXT	off	Select graphics modes
$C051	R/W	TEXT	on	Select text modes
$C052	R/W	MIX	off	Select non-mixed graphics
$C053	R/W	MIX	on	Select mixed text/ graphics
$C054	R/W	DPAGE2	off	Select display page 1
$C055	R/W	DPAGE2	on	Select display page 2
$C056	R/W	HGR	off	Select low-resolution graphics
$C057	R/W	HGR	on	Select high-resolution graphics
$C05E	R/W	INHDRGR	off	enable double-resolution graphics
$C05F	R/W	INHDRGR	on	Inhibit double-resolution graphics

Table 5-1 Video display control switches I/O locations

Display mode	Control switch setting				
	DBLRES	TEXT	MIX	HGR	INHDRGR
40-column text	off	on	-	-	-
80-column text	on	on	-	-	-
Low-res graphics	off	off	off	off	-
	-	off	off	off	on
Dbl-low-res graphics	on	off	off	off	off
Hi-res graphics	off	off	off	on	-
	-	off	off	on	on
Dbl-hi-res graphics	on	off	off	on	off
Low-res/40-column text	off	off	on	off	-
Low-res/80-column text	on	off	on	off	on
Dbl-low-res/80-column text	on	off	on	off	off
Hi-res/40-column text	off	off	on	on	-
Hi-res/80-column text	on	off	on	on	on
Dbl-hi-res/80-column text	on	off	on	on	off

Table 5-2 Selection of video display mode

Display page	Control switch setting	
	DPAGE2	INHPAGE2
1	off	-
	-	on
2	on	off

Table 5-3 Selection of display page

The video display of the computer is memory-mapped, i.e. part of the system main memory is reserved for video display purposes. Each of the video memory locations is "mapped" to a particular drawing position on the screen. Depending on the video display mode, the resolution and color of the display, memory requirement, memory mapping and interpretation of video memory data may be different. These will be described in detail in the following sections.

5.1 Text modes

Text modes are selected by turning the "TEXT" switch on. In both the 40-column and 80-column text mode, the screen can display 24 rows of text. Each of the characters occupies a 7H x 8V dot-matrix on the screen. Except for some special characters, most of the characters are actually made up of a 5H x 7V dot-matrix, leaving a blank dot column at both sides and a blank dot row at the bottom of the character. Figure 5-1 shows the dot-matrix of the character "C".

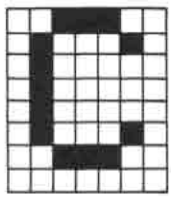


Fig. 5-1 Dot-matrix of character "C"

5.1.1 Character sets

The characters that can be displayed in the text modes include:

- Uppercase and lowercase letters
- Numerals
- Punctuation marks
- Special characters

An unique 8-bit character code is assigned to each of the displayable characters. The low-order six bits are simply the ASCII code for the character while the two high-order bits select one of three character display formats:

- Normal (white character on a black background)
- Inverse (black character on a white background)
- Flashing (alternating between normal and inverse format)

The computer has two character sets selected by the "CHARSET2" switch.

When "CHARSET2" switch is off, the primary character set is displayed. Uppercase letters, numerals and punctuation marks can be displayed in all three formats while lowercase letters and some special characters can only be displayed in normal format. The primary character set and the corresponding character codes are shown in Table 5-4.

	INVERSE				FLASHING				NORMAL							
	$00	$10	$20	$30	$40	$50	$60	$70	$80	$90	$A0	$B0	$C0	$D0	$E0	$F0
+$0	@	P		0	@	P		0	@	P		0	@	P	`	p
+$1	A	Q	!	1	A	Q	!	1	A	Q	!	1	A	Q	a	q
+$2	B	R	"	2	B	R	"	2	B	R	"	2	B	R	b	r
+$3	C	S	#	3	C	S	#	3	C	S	#	3	C	S	c	s
+$4	D	T	$	4	D	T	$	4	D	T	$	4	D	T	d	t
+$5	E	U	%	5	E	U	%	5	E	U	%	5	E	U	e	u
+$6	F	V	&	6	F	V	&	6	F	V	&	6	F	V	f	v
+$7	G	W	'	7	G	W	'	7	G	W	'	7	G	W	g	w
+$8	H	X	(	8	H	X	(	8	H	X	(	8	H	X	h	x
+$9	I	Y	)	9	I	Y	)	9	I	Y	)	9	I	Y	i	y
+$A	J	Z	*	:	J	Z	*	:	J	Z	*	:	J	Z	j	z
+$B	K	[	+	;	K	[	+	;	K	[	+	;	K	[	k	{
+$C	L	\	,	<	L	\	,	<	L	\	,	<	L	\	l	|
+$D	M	]	-	=	M	]	-	=	M	]	-	=	M	]	m	}
+$E	N	^	.	>	N	^	.	>	N	^	.	>	N	^	n	~
+$F	O	_	/	?	O	_	/	?	O	_	/	?	O	_	o	▒

Table 5-4 Primary character set

When "CHARSET2" switch is on, the secondary character set is displayed. Flashing format is not available for this character set but lowercase letters can now be displayed in inverse format. Moreover, some special characters (called mouse characters) can be displayed. Table 5-5 shows the secondary character set and the corresponding character codes.

	INVERSE				MOUSE		INVERSE		NORMAL							
	$00	$10	$20	$30	$40	$50	$60	$70	$80	$90	$A0	$B0	$C0	$D0	$E0	$F0
+$0	@	P		0	▲		`	p	@	P		0	@	P	`	p
+$1	A	Q	!	1	△		a	q	A	Q	!	1	A	Q	a	q
+$2	B	R	"	2	▲		b	r	B	R	"	2	B	R	b	r
+$3	C	S	#	3	◇		c	s	C	S	#	3	C	S	c	s
+$4	D	T	$	4	✓		d	t	D	T	$	4	D	T	d	t
+$5	E	U	%	5	◪		e	u	E	U	%	5	E	U	e	u
+$6	F	V	&	6	→		f	v	F	V	&	6	F	V	f	v
+$7	G	W	'	7	▒		g	w	G	W	'	7	G	W	g	w
+$8	H	X	(	8	↑		h	x	H	X	(	8	H	X	h	x
+$9	I	Y	)	9	▒		i	y	I	Y	)	9	I	Y	i	y
+$A	J	Z	*	:	↓		j	z	J	Z	*	:	J	Z	j	z
+$B	K	[	+	;	↑		k		K	[	+	;	K	[	k	
+$C	L	\	,	<	▒		l		L	\	,	<	L	\	l	
+$D	M	]	-	=	▒		m	]	M	]	-	=	M	]	m	]
+$E	N	^	.	>	▒		n	~	N	^	.	>	N	^	n	~
+$F	O	_	/	?	▒		o	▒	O	_	/	?	O	_	o	▒

Table 5-5 Secondary character set

5.1.2 Memory mapping

In both of the text modes, each character position on the screen is mapped to an unique location in the system memory. To display a specific character at a particular position on the screen, you only need to store the character code for the character to be displayed in the memory location which is mapped to the desired character position on the screen.

5.1.2.1 40-column text

40-column text mode is selected by turning the "DBLRES" switch off. In this mode, the screen is divided into 40 x 24 character positions. The 40 characters in a row are stored in contiguous memory locations called a row buffer. However, the row buffers for adjacent rows on the screen are not adjacent in the system memory.
The primary 40-column text page occupies locations $0400 to $07FF while the secondary page occupies locations $0800 to $0BFF of the main bank system memory. The memory mapping is illustrated in Fig. 5-2.

Base address		Offset			
Page 1	Page 2	$00 $27	$28 $4F	$50 $77	$78 $7F
$0400	$0800	Row 0	Row 8	Row 16	Reserved
$0480	$0880	Row 1	Row 9	Row 17	Reserved
$0500	$0900	Row 2	Row 10	Row 18	Reserved
$0580	$0980	Row 3	Row 11	Row 19	Reserved
$0600	$0A00	Row 4	Row 12	Row 20	Reserved
$0680	$0A80	Row 5	Row 13	Row 21	Reserved
$0700	$0B00	Row 6	Row 14	Row 22	Reserved
$0780	$0B80	Row 7	Row 15	Row 23	Reserved

Fig. 5-2 40-column text mode memory mapping

Notice that some memory locations inside both of the display pages are marked as "reserved". They are sometimes referred to as "screen holes" since they are not used for display purposes, i.e. they are not mapped to any position on the screen.

These "screen holes" are used by the system firmware and some application programs for data storage. As a result, the user should pay particular attention when using these memory locations in order to avoid destroying any data which may cause system failure.

5.1.2.2 80-column text

80-column text mode is selected by turning "DBLRES" switch on. In this mode, the screen is divided into 80 x 24 character positions.

The primary 80-column text page occupies locations $400 to $7FF while the secondary page occupies locations $800 to $BFF of both the main and auxiliary bank system memory. Thus a 80-column text page uses twice as much memory as a 40-column text page.

The row buffer for a particular row on the screen uses the same set of memory addresses as 40-column text. However, both the main and auxiliary bank memory is used and adjacent character positions in a row on the screen are not mapped to contiguous memory locations in system memory.

If we label the character positions in a row on the screen as 0,1,2,....,78,79 (starting from the leftmost position), then the even-number character positions (0,2,4,....,76,78) are mapped to the auxiliary bank system memory while the odd-number character positions (1,3,5,....77,79) are mapped to the same memory addresses in the main bank system memory. For example, the first character position of the first row on the

screen is mapped to location $0400 (or $0800 if secondary page is being displayed) of auxiliary bank system memory while the second character position is mapped to location $0400 ($0800 for secondary page) of main bank system memory. Figure 5-3 illustrates this mapping scheme conceptually.

Base address		Offset								Row
Page 1	Page 2	$00	$00	$01	$01	$02	$02	. . . .	$27	$27	
$0400	$0800	A	M	A	M	A	M	. . . .	A	M	0
$0480	$0880	A	M	A	M	A	M	. . . .	A	M	1
$0500	$0900	A	M	A	M	A	M	. . . .	A	M	2
.	.	.	.	.	.	.		. . . .	.	.	.
.	.	.	.	.	.	.		. . . .	.	.	.
.	.	.	.	.	.	.	.	. . . .	.	.	.
$0780	$0B80	A	M	A	M	A	M	. . . .	A	M	7
$428	$0828	A	M	A	M	A	M	. . . .	A	M	8
.	.	.	.	.	.	.	.	. . . .	.	.	.
.	.	.	.	.	.	.	.	. . . .	.	.	.
.	.	.	.	.	.	.	.	. . . .	.	.	.
$07A8	$0BA8	A	M	A	M	A	M	. . . .	A	M	15
$0450	$0850	A	M	A	M	A	M	. . . .	A	M	16
.	.	.	.	.	.	.	.	. . . .	.	.	.
.	.	.	.	.	.	.	.	. . . .	.	.	.
.	.	.	.	.	.	.	.	. . . .	.	.	.
$07D0	$0BD0	A	M	A	M	A	M	. . . .	A	M	23
Column		0	1	2	3	4	5	. . . .	78	79	

A - Auxiliary bank memory
M - Main bank memory

Fig. 5-3 80-column text mode memory mapping

5.2 Graphics modes

Graphics modes are selected by turning the switches "TEXT" and "MIX" off.

In all of the graphics modes, the entire screen is divided into a rectangular pixel-matrix. Each of the pixels on the screen is mapped to one or more bits of a particular location in the system memory.

The resolution, color and memory mapping are different for the various graphics modes and will be described in the following sections.

By the way, we made a distinction between a "dot" and a "pixel" here.

A "dot" is the smallest element that can be drawn on the screen. The entire screen is made up of 560H x 192V drawing positions at which a "dot" can be plotted.

On the other hand, a "pixel" is the basic picture element of a particular graphics mode which may be made up of one or more dots. Thus the size of a "pixel" may be equal to or larger than that of a "dot". Obviously, the smaller the pixel size, the higher the resolution.

5.2.1 Low-resolution graphics

This is selected by turning off "HGR" switch and either turning off "DBLRES" switch or turning on "INHDRGR" switch. In this mode, the screen is divided into 40H x 48V pixels. Each pixel can take on any one of 16 colors.

It uses the same memory area as 40-column text, i.e. $0400 to $07FF (primary page) and $800 to $0BFF (secondary page) of the main bank system memory.

The mapping scheme is also similar to that of 40-column text except that each character position is now divided vertically into two pixel positions (upper and lower) which are mapped to the low-nibble and high-nibble of the memory location mapped to that particular character position on the screen. The mapping scheme is shown in Fig. 5-4.

Base address		Offset									
Page 1	Page 2	$00	$01	$02	$03	. . . .	$24	$25	$26	$27	Row
$0400	$0800	L	L	L	L	. . . .	L	L	L	L	0
$0400	$0800	H	H	H	H	. . . .	H	H	H	H	1
$0480	$0880	L	L	L	L	. . . .	L	L	L	L	2
$0480	$0880	H	H	H	H	. . . .	H	H	H	H	3
.	.	.	.	.	.	. . . .	.	.	.	.	.
.	.	.	.	.	.	. . . .	.	.	.	.	.
.	.	.	.	.	.	. . . .	.	.	.	.	.
.	.	.	.	.	.	. . . .	.	.	.	.	.
$0750	$0B50	L	L	L	L	. . . .	L	L	L	L	44
$0750	$0B50	H	H	H	H	. . . .	H	H	H	H	45
$07D0	$0BD0	L	L	L	L	. . . .	L	L	L	L	46
$07D0	$0BD0	H	H	H	H	. . . .	H	H	H	H	47
Column		0	1	2	3	. . . .	36	37	38	39	

L - Low-nibble
H - High-nibble

Fig. 5-4 Low-resolution graphics memory mapping

To set the color of a particular pixel position on the screen, you only need to store the 4-bit color code for the desired color in the low-nibble or high-nibble (as appropriate) of the memory location which is mapped to that pixel position. Table 5-6 shows the 16 available colors and their corresponding color codes.

Code	Color	Code	Color
$0	Black	$8	Yellowish-green
$1	Dark red	$9	Orange
$2	Dark blue	$A	Gray 2
$3	Violet	$B	Pink
$4	Dark green	$C	Medium green
$5	Gray 1	$D	Yellow
$6	Medium blue	$E	Cyan
$7	Light blue	$F	White

Table 5-6 Low-resolution graphics colors

5.2.2 Double-low-resolution graphics

This mode is selected by turning off "HGR" and "INHDRGR" switch and turning on "DBLRES" switch.

In this mode, the screen is divided into 80H x 48V pixel positions, each of which can take on any one of the 16 colors shown in Table 5-7.

Color	Code	
	Even column	Odd column
Black	$0	$0
Dark Red	$8	$1
Dark Blue	$1	$2
Violet	$9	$3
Dark green	$2	$4
Gray 1	$A	$5
Medium blue	$3	$6
Light blue	$B	$7
Yellowish-green	$4	$8
Orange	$C	$9
Gray 2	$5	$A
Pink	$D	$B
Medium green	$6	$C
Yellow	$E	$D
Cyan	$7	$E
White	$F	$F

Table 5-7 Double low-resolution graphics colors

Notice that the 4-bit color code of a particular color for pixels on even columns (0,2,4,....,78) of the screen is different from that for pixels on odd columns (1,3,5,....,79).

An observant reader may recognize that the color codes for odd-column pixels are the same as those of low-resolution graphics while the even-column color codes are obtained by rotating their odd-column counterparts one bit to the right. The memory mapping is similar to that of 80-column text except that, as in low-resolution graphics, each character position is divided vertically into two pixel positions mapped to the low-nibble and high-nibble of the memory location mapped to that character position.

The primary display page occupies locations $0400 to $07FF and the secondary display page occupies locations $0800 to $0BFF of both the main and auxiliary bank system memory. Odd-column pixels are mapped to main bank memory while even-column pixels are mapped to auxiliary bank memory.

Even-row pixels are mapped to low-nibbles while odd-row pixels are mapped to high-nibbles of the memory locations in the display buffers.

The mapping scheme is illustrated in Fig. 5-5.

Base address		Offset								Row
Page 1	Page 2	$00	$00	$01	$01	$02	$02	$27	$27	
$0400	$0800	AL	ML	AL	ML	AL	ML	AL	ML	0
$0400	$0800	AH	MH	AH	MH	AH	MH	AH	MH	1
$0480	$0880	AL	ML	AL	ML	AL	ML	AL	ML	2
$0480	$0880	AH	MH	AH	MH	AH	MH	AH	MH	3
$0500	$0900	AL	ML	AL	ML	AL	ML	AL	ML	4
.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.
$06D0	$0AD0	AH	MH	AH	MH	AH	MH	AH	MH	43
$0750	$0B50	AL	ML	AL	ML	AL	ML	AL	ML	44
$0750	$0B50	AH	MH	AH	MH	AH	MH	AH	MH	45
$07D0	$0BD0	AL	ML	AL	ML	AL	ML	AL	ML	46
$07D0	$0BD0	AH	MH	AH	MH	AH	MH	AH	MH	47
Column		0	1	2	3	4	5	78	79	

ML-Low-nibble of main memory
MH-High-nibble of main memory
AL-Low-nibble of auxiliary memory
AH-High-nibble of auxiliary memory

Fig. 5-5 Double-low-resolution graphics memory mapping

5.2.3 High-resolution graphics

High-resolution graphics screen contains 280H x 192V plotting positions. This mode is selected by turning on "HGR" switch and either turning off "DBLRES" switch or turning on "INHDRGR" switch.

There are two submodes, namely monochrome and color, selected by the "COLOR/MONO" switch located on the top cabinet.

5.2.3.1 Monochrome high-resolution graphics

The monochrome submode, selected by throwing the "COLOR/MONO" switch to the "MONO" position, is for use with a high-resolution monochrome monitor.

The primary display page occupies locations $2000 to $3FFF while the secondary display page occupies locations $4000 to $5FFF of the main bank system memory.

Each of the display buffers is organized as a collection of line buffers. Each line buffer is made up of 40 contiguous memory locations and is mapped to one of the 192 lines on the screen. Line buffers for adjacent lines on the screen are not adjacent in memory. The organisation of the display buffers is shown in Fig. 5-6.

Bit 0 to bit 6 of each memory location in a line buffer is mapped to seven adjacent plotting positions on the associated line, with bit 0 mapped to the left and bit 6 mapped to the right. Bit 7 is not used.

Bit 6 of a particular memory location in a line buffer and bit 0 of the next sequential location in the buffer are mapped to adjacent plotting positions on the screen. The bit-mapping scheme is illustrated in Fig. 5-7.

In the monochrome mode, each pixel occupies one plotting position on the screen and can either be black or white only. Color is not available.

To set a particular plotting position to white, store a one to the memory bit mapped to that position. Storing a Zero set the corresponding plotting position to black.

Page1	Page2	$0 $27	$28 $4F	$50 $77	$78 $7F	$80 $A7	$A8 $CF	$D0 $F7	$F8 $FF
$2000	$4000	line 0	line 64	line 128	reserved	line 8	line 72	line 136	reserved
$2100	$4100	line 16	line 80	line 144	reserved	line 24	line 88	line 152	reserved
$2200	$4200	line 32	line 96	line 160	reserved	line 40	line 104	line 168	reserved
$2300	$4300	line 48	line 112	line 176	reserved	line 56	line 120	line 184	reserved
$2400	$4400	line 1	line 65	line 129	reserved	line 9	line 73	line 137	reserved
$2500	$4500	line 17	line 81	line 145	reserved	line 25	line 89	line 153	reserved
$2600	$4600	line 33	line 97	line 161	reserved	line 41	line 105	line 169	reserved
$2700	$4700	line 49	line 113	line 177	reserved	line 57	line 121	line 185	reserved
$2800	$4800	line 2	line 66	line 130	reserved	line 10	line 74	line 138	reserved
$2900	$4900	line 18	line 82	line 146	reserved	line 26	line 90	line 154	reserved
$2A00	$4A00	line 34	line 98	line 162	reserved	line 42	line 106	line 170	reserved
$2B00	$4B00	line 50	line 114	line 178	reserved	line 58	line 122	line 186	reserved
$2C00	$4C00	line 3	line 67	line 131	reserved	line 11	line 75	line 139	reserved
$2D00	$4D00	line 19	line 83	line 147	reserved	line 27	line 91	line 155	reserved
$2E00	$4E00	line 35	line 99	line 163	reserved	line 43	line 107	line 171	reserved
$2F00	$4F00	line 51	line 115	line 179	reserved	line 59	line 123	line 187	reserved
$3000	$5000	line 4	line 68	line 132	reserved	line 12	line 76	line 140	reserved
$3100	$5100	line 20	line 84	line 148	reserved	line 28	line 92	line 156	reserved
$3200	$5200	line 36	line 100	line 164	reserved	line 44	line 108	line 172	reserved
$3300	$5300	line 52	line 116	line 180	reserved	line 60	line 124	line 188	reserved
$3400	$5400	line 5	line 69	line 133	reserved	line 13	line 77	line 141	reserved
$3500	$5500	line 21	line 85	line 149	reserved	line 29	line 93	line 157	reserved
$3600	$5600	line 37	line 101	line 165	reserved	line 45	line 109	line 173	reserved
$3700	$5700	line 53	line 117	line 181	reserved	line 61	line 125	line 189	reserved
$3800	$5800	line 6	line 70	line 134	reserved	line 14	line 78	line 142	reserved
$3900	$5900	line 22	line 86	line 150	reserved	line 30	line 94	line 158	reserved
$3A00	$5A00	line 38	line 102	line 166	reserved	line 46	line 110	line 174	reserved
$3B00	$5B00	line 54	line 118	line 182	reserved	line 62	line 126	line 190	reserved
$3C00	$5C00	line 7	line 71	line 135	reserved	line 15	line 79	line 143	reserved
$3D00	$5D00	line 23	line 87	line 151	reserved	line 31	line 95	line 159	reserved
$3E00	$5E00	line 39	line 103	line 167	reserved	line 47	line 111	line 175	reserved
$3F00	$5F00	line 55	line 119	line 183	reserved	line 63	line 127	line 191	reserved

Fig. 5-6 Display buffer organisation in high-resolution graphics

Base address		Offset												
Page 1	Page 2	$00	$00	$00	$00	$00	$00	$00	$01	$01	. . .	$27	$27	Row
$2000	$4000	B0	B1	B2	B3	B4	B5	B6	B0	B1	. . .	B5	B6	0
$2400	$4400	B0	B1	B2	B3	B4	B5	B6	B0	B1	. . .	B5	B6	1
$2800	$4400	B0	B1	B2	B3	B4	B5	B6	B0	B1	. . .	B5	B6	2
$2C00	$4C00	B0	B1	B2	B3	B4	B5	B6	B0	B1	. . .	B5	B6	3
.	.	.	.	.	.	.	.	.	.	.	. . .	.	.	.
.	.	.	.	.	.	.	.	.	.	.	. . .	.	.	.
.	.	.	.	.	.	.	.	.	.	.	. . .	.	.	.
$33D0	$53D0	B0	B1	B2	B3	B4	B5	B6	B0	B1	. . .	B5	B6	188
$37D0	$57D0	B0	B1	B2	B3	B4	B5	B6	B0	B1	. . .	B5	B6	189
$3BD0	$5BD0	B0	B1	B2	B3	B4	B5	B6	B0	B1	. . .	B5	B6	190
$3FD0	$5FD0	B0	B1	B2	B3	B4	B5	B6	B0	B1	. . .	B5	B6	191
Column		0	1	2	3	4	5	6	7	8	. . .	278	279	

Fig. 5-7 Bit-mapping of high-resolution graphics.

5.2.3.2 Color high-resolution graphics

The color submode selected by throwing the "COLOR/MONO" switch to the "COLOR" position is for use with a color TV or color monitor.

In this mode, four colors can be displayed on the screen in addition to black and white. If viewed with a composite monochrome monitor, the different colors will be displayed as different gray levels.

The memory mapping is the same as that of the monochrome submode except that the video memory data is interpreted in a different manner in order to generate a color display. The color of a particular plotting position on the screen is determined by five factors:

- The plotting position is on even columns (0,2,4,....,278) or odd columns (1,3,5,...,279) of the screen.

- The state of the memory bit mapped to that plotting position.

- The state of bit 7 of the memory location which contains the bit mapped to that plotting position.

- The state of the two memory bits mapped to the left of the plotting position.

- The state of the memory bit mapped to the right of the plotting position.

A plotting position will be white if the memory bits mapped to the current plotting position and either one of the adjacent (left or right) plotting positions on the same line are on.

A plotting position will be black if the memory bit mapped to the current plotting position is off and either the bit mapped to the position at the left is off or that plotting position is white in color.

A plotting position can only be displayed in colors other than black or white if the memory bit mapped to it is on and the bits mapped to the two adjacent plotting positions are off. The plotting position at the right will be forced to the same color.

By the way, the horizontal border area, i.e. the left of column 0 and the right of column 279, will always be black in color. The memory bits mapped to this area can be treated as zero's.

The color encoding scheme is illustrate in Table 5-8.

Column	State of memory bits					Color
	S0	S1	S2	S3	B7	
X	X	0	0	X	X	Black
X	1	1	0	X	X	Black
X	X	1	1	X	X	White
X	X	X	1	1	X	White
Even	X	0	1	0	0	Violet
Even	X	0	1	0	1	Medium blue
Odd	X	0	1	0	0	Medium green
Odd	X	0	1	0	1	Orange
X	0	1	0	X	X	Unchanged

S0 : Two plotting positions to the left
S1 : One plotting position to the left
S2 : Current plotting position
S3 : One plotting position to the right
B7 : Bit 7 of location mapped to current plotting position
X : Don't care (0 or 1)

Table 5-8 High-resolution graphics color encoding scheme

Using this color encoding scheme, each color pixel (except black) occupies two or more plotting positions on the screen so that the effective horizontal color resolution is 140 instead of 280 as in the monochrome submode.

This is to be expected since with the same amount of display memory, there is always a trade-off between higher resolution and more colors.

5.2.4 Double-high-resolution graphics

Double-high-resolution graphics screen is divided into 560H x 192V plotting positions. It is selected by turning on "HGR" and "DBLRES" and turning off "INHDRGR".

As in high-resolution graphics, there are two submodes (monochrome and color) selected by the "MONO/COLOR" switch on the top cabinet of the computer.

5.2.4.1 Monochrome double-high-resolution graphics

The monochrome submode is selected by throwing the "MONO/COLOR" switch to the "MONO" position.

In this mode, each pixel occupies one plotting position on the screen and can either be black or white only. Color is not available.

The display buffers occupy locations $2000 to $3FFF (page 1) and $4000 to $5FFF (page 2) of both the main and auxiliary bank memory. Thus double-high-resolution graphics uses twice as much memory as high-resolution graphics.

The organisation of line buffers is similar to that of high-resolution graphics, except that each line buffer is made up of 40 contiguous memory locations in the main bank memory and the 40 memory locations in the auxiliary bank memory having the same addresses.

As in high-resolution graphics, each memory location in a line buffer is mapped to seven adjacent plotting positions on the screen, with bit 0 mapped to the left and bit 6 mapped to the right. Bit 7 is not used.

However, data in adjacent memory locations in either the main or auxiliary bank memory are not displayed side-by-side on the screen. Instead, the data in main and auxiliary bank memory are displayed alternately, i.e. one byte in the auxiliary bank memory is displayed first, followed by another byte in the main bank memory and so on.

If you conceptually divide a line on the screen into 80 segments, each of which is seven plotting positions in width, then the even-number segments (0,2,4,...,78) are mapped to the 40 contiguous memory locations of the line buffer in the auxiliary bank memory while the odd number segments (1,3,5,...,79) are mapped to the same addresses in the main bank memory.

To set a particular plotting position on the screen to white, you simply need to place a one into the memory bit which is mapped to that plotting position. Placing a zero in the memory bit set the corresponding plotting position to black.

The bit-mapping scheme is illustrated in Fig. 5-8.

Base address		Offset															
Page 1	Page 2	$00	$00	. . .	$00	$00	$00	. . .	$00	$01	$01	. . .	$27	$27	Row
$2000	$4000	A0	A1	. . .	A6	M0	M1	. . .	M6	A0	A1	. . .	M5	M6	0
$2400	$4400	A0	A1	. . .	A6	M0	M1	. . .	M6	A0	A1	. . .	M5	M6	1
$2800	$4800	A0	A1	. . .	A6	M0	M1	. . .	M6	A0	A1	. . .	M5	M6	2
$2C00	$4C00	A0	A1	. . .	A6	M0	M1	. . .	M6	A0	A1	. . .	M5	M6	3
.	.	.	.	. . .	.	.	.	. . .	.	.	.	. . .	.	.	.
.	.	.	.	. . .	.	.	.	. . .	.	.	.	. . .	.	.	.
.	.	.	.	. . .	.	.	.	. . .	.	.	.	. . .	.	.	.
$33D0	$53D0	A0	A1	. . .	A6	M0	M1	. . .	M6	A0	A1	. . .	M5	M6	188
$37D0	$37D0	A0	A1	. . .	A6	M0	M1	. . .	M6	A0	A1	. . .	M5	M6	189
$3BD0	$5BD0	A0	A1	. . .	A6	M0	M1	. . .	M6	A0	A1	. . .	M5	M6	190
$3FD0	$5FD0	A0	A1	. . .	A6	M0	M1	. . .	M6	A0	A1	. . .	M5	M6	191
Column		0	1	. . .	6	7	8	. . .	13	14	15	. . .	558	559	

A0-A6 : Auxiliary bank memory bit 0 to bit 6

M0-M6 : Main bank memory bit 0 to bit 6

Fig. 5-8 Double-high-resolution graphics bit-mapping

5.2.4.2 Color double-high-resolution graphics

The color submode selected by throwing the "MONO/COLOR" switch to the "COLOR" position is intended to be used with a color monitor or TV.

It has the same memory mapping as the monochrome submode. However, the video data addressed from the display buffer is interpreted as specially encoded color data instead of a simple bit pattern.

The color of a particular plotting position on the screen is determined by the following factors:

- The column at which the plotting position locates (0-559).
- The state of the memory bit mapped to that plotting position.
- The state of the memory bits mapped to the four plotting positions at the left.
- The state of the memory bits mapped to the three plotting positions at the right.

If the memory bits mapped to four or more adjacent plotting positions on a line are on, then all of the plotting positions will be white.

If the memory bit mapped to a particular plotting position is on and the bit mapped to the plotting position at the left is off, then it will take on different colors depending on the state of the memory bits mapped to the three plotting positions at the right and the column number at which it locates as shown in table 5-9.

One to three plotting positions at the right will be forced to the same color as that of the current position. Thus the width of a pixel (other than black and white) varies from two to four plotting positions so that the effective horizontal color resolution is only 140.

Bit pattern					C MOD 4				
B0	B1	B2	B3	B4	0	1	2	3	Width
0	1	0	0	0	Dark blue	Dark green	Yelowish-green	Dark red	4
0	1	0	0	1	Violet	Medium blue	Medium green	Orange	3
0	1	0	1	0	Gray 2	Gray 1	Gray 2	Gray 1	2
0	1	0	1	1	Pink	Light blue	Cyan	Yellow	2
0	1	1	0	0	Medium blue	Medium green	Orange	Violet	4
0	1	1	0	1	Light blue	Cyan	Yellow	Pink	3
0	1	1	1	0	Cyan	Yellow	Pink	Light blue	4

C : Column number of current plotting position (0 to 559)

B0 : Bit mapped to one plotting position to the left

B1 : Bit mapped to current plotting position

B2 : Bit mapped to one plotting position to the right

B3 : Bit mapped to two plotting positions to the right

B4 : Bit mapped to three plotting positions to the right

Table 5-9 Double-high-resolution graphics color encoding scheme

As in high-resolution graphics, the horizontal border area can be treated as if they are mapped to memory bits in the off state.

If the memory bit mapped to a particular plotting position and those mapped to the three plotting positions at the left are all off, then that plotting position will be black. However, if one or more bits mapped to the three plotting positions at the left is on, then it will take on the color of the previous plotting position except when the previous plotting position is white, in which case the current plotting position will be black.

Like color high-resolution graphics, the color double-high-resolution graphics scarifies resolution for colors. The horizontal resolution is reduced from 560 to 140 in order to increase the available colors from two to 16.

When viewed with a composite monochrome monitor, the different colors will appear as different levels of gray.

5.3 Mixed graphics/text modes

Mixed modes are selected by turning "TEXT" switch off and "MIX" switch on. In all of the mixed modes, the screen is divided vertically into two sections.

The upper section is a graphics display, the particular graphics mode being selected by the software-switches "HGR", "DBLRES" and "INHDRGR" as described in section 5.2. The lower section is a text display which contains four rows of text. The text may be in 40-column or 80-column format, selected by the "DBLRES" software-switch.

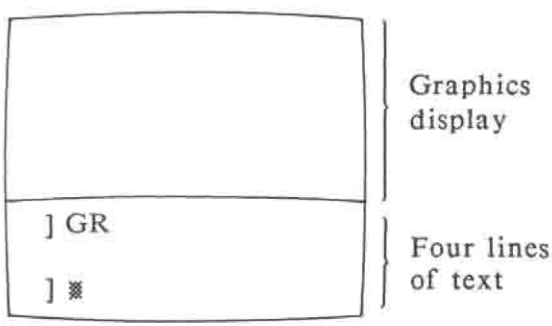


Fig. 5-9 Mixed graphics/text display

Notice that since double-low-resolution graphics and double-high-resolution graphics require the "DBLRES" software-switch to be turned on, they can only be mixed with 80-column text. Low-resolution graphics and high-resolution graphics can be mixed with either 40-column or 80-column text.

For the two low-resolution graphics modes, the graphics screen can only display 40 lines. For the two high-resolution graphics modes, 160 lines can be displayed.

5.4 Secondary display page

For each of the display modes, there are two display pages or display buffers available. Some display modes may share the same display buffers, e.g. 40-column text and low-resolution graphics. The display page is selected by the software-switches "DPAGE2" and "INHPAGE2" as shown in Table 5-3.

When "INHPAGE2" is off, "DPAGE2" is used for selecting the display page. If "DPAGE2" is off, display page 1 is selected. If "DPAGE2" is on, display page 2 is selected. If "INHPAGE2" is turned on, display page 1 is always selected irrespective of the state of the "DPAGE2" switch. "DPAGE2" is used for other purposes (switching between main and auxiliary bank display buffers) as described in Chapter 3.

With this secondary display page feature and the ability to monitor the state of the vertical blanking signal by reading bit 7 of I/O location $C019, it is possible to perform flicker-free animation.

To produce a flicker-free display, you must not write to the display buffer during the active display interval when the video generation circuitry is accessing the contents of the video memory.

With only one display buffer, it is possible to achieve this by monitoring the vertical blanking signal and writing to the display buffer during the blanking interval. However if the amount of data to be handled is large, there may not be enough time for updating the display buffer during the blanking interval.

With two display buffers, you may modify the content of one of the display buffers while displaying the other and switch to it during the vertical blanking interval after you have finished writing. Since the entire display is switched at the same time, no flickering will be observed.

As a final remark, when you switch the display page of a mixed mode display, both the graphics section and the text section will be switched at the same time. It is not possible to display graphics page 1 and text page 2 on the screen at the same time.

CHAPTER 6
DISK INPUT/OUTPUT

6. DISK INPUT/OUTPUT

The computer has all the necessary hardware and firmware built-in for interfacing with 5.25" and 3.5" disk drives. The following types of disk drives are supported:

- LASER FD-100c 5.25" drives or compatibles
- Unidisk™ 5.25 drives or compatibles
- LASER FD-356 3.5" drives or compatibles
- Unidisk™ 3.5 drives or compatibles

The "LASER 128" and "LASER 128 EX" has a 5.25" disk drive built-in.

The DB-19 connector on the back panel of the computer is for connecting an optional second drive.

6.1 Basic operating principles of disk drives

A disk drive is an assembly of electrical and mechanical parts for reliable data storage and retrieval on a magnetic media, i.e. the diskette.

Basically, a disk drive is composed of a spindle motor, a stepper motor, a magnetic read-write head, electronic components for controlling the motors and read/write head and other mechanisms for mounting the diskette, moving and loading the magnetic head etc.. Fig. 6-1 shows the construction of a typical drive assembly.

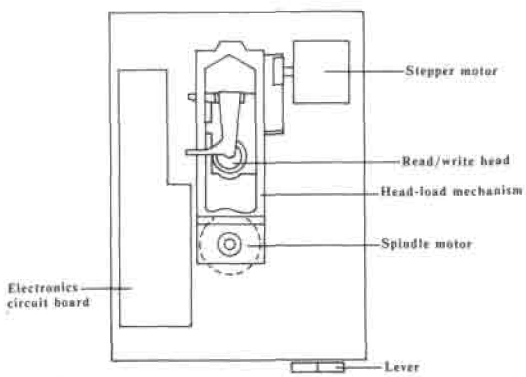


Fig. 6-1 Basic construction of a disk drive

When a diskette is inserted into the disk drive and the drive door is closed properly, the diskette will be mounted firmly on the drive hub which is connected to the shaft of the spindle motor, either directly or through a rubber belt. If the spindle motor is then turned on, the diskette will revolve at high speed about its center axis.

The read-write head is held above the surface of the diskette by a disk arm which is connected to the shaft of the stepper motor through some mechanisms. When the drive door is closed, the read-write head will be "loaded", i.e. brought into contact with the surface of the diskette.

The stepper motor is responsible for accurately controlling the position of the read/write head. The read/write head can be moved inwards (towards the centre) or outwards (away from the centre) in fixed "steps".

Data are stored on concentric "tracks" on the diskette. The outermost track is often referred to as track 0. The number of tracks depends on the accuracy of the magnetic head positioning mechanism and the quality of the stepper motor which may be different for different disk drives.

The magnetic read/write head is the component which is responsible for actually storing and retrieving data on the diskette.

When writing to the diskette, it converts the one's and zero's in the serial bit stream from the host computer into magnetic flux changes which magnetize the magnetic media coated on the surface of the diskette in opposite poles. When the disk surface moves at high speed under the read/write head, the magnetic flux changes recorded on the track induce voltage spikes in the read/write head. These voltage spikes are amplified and shaped by the electronics in the disk drive and passed to the host computer for reading.

For more effective error detection, a track is usually divided into a number of sectors. Some disk drives use "hard-sectoring" while others use "soft-sectoring". All the disk drives which work with the computer uses the "soft-sectoring" scheme.

In this scheme, each data sector is preceded by an address mark which uniquely identifies the current track and sector. The number of sectors in a track and the number of bytes in a sector is defined by software, typically 16 sectors/track. The larger the number of sectors, the smaller the number of bytes per sector.

Usually, each sector has a data checksum for error detection. For efficient error detection and recovery, smaller sector size is preferred. However, this also increases the overhead spent in storing the address marks, checksums, inter-sector gap etc. which results in reduced useful storage space per track. This tradeoff is another consideration which a programmer has to bear in mind.

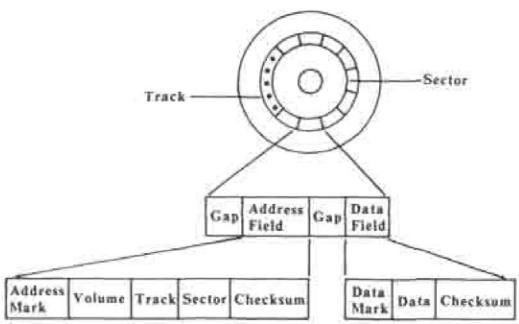


Fig. 6-2 Typical format of a soft-sectored diskette

6.2 Universal Disk Controller

The disk drive interface hardware in the computer is collectively known as the Universal Disk Controller (UDC). The UDC uses a number of memory mapped I/O locations for controlling the operation of the disk drives. The locations and functions of these I/O locations are shown in Table 6-1 and Table 6-2.

Address	Operation	Switch/ Register	State	Function
$C0E0	R/W	PHASE0	Off	Stepper motor phase 0 off
$C0E1	R/W	PHASE0	On	Stepper motor phase 0 on
$C0E2	R/W	PHASE1	Off	Stepper motor phase 1 off
$C0E3	R/W	PHASE1	On	Stepper motor phase 1 on
$C0E4	R/W	PHASE2	Off	Stepper motor phase 2 off
$C0E5	R/W	PHASE2	On	Stepper motor phase 2 on
$C0E6	R/W	PHASE3	Off	Stepper motor phase 3 off
$C0E7	R/W	PHASE3	On	Stepper motor phase 3 on
$C0E8	R/W	ENABLE	Off	Disable disk drive
$C0E9	R/W	ENABLE	On	Enable disk drive
$C0EA	R/W	DRIVE2	Off	Select first drive
$C0EB	R/W	DRIVE2	On	Select second drive
$C0EC	R/W	SENSE	Off	WRITE off: read disk data
$C0ED	R/W	SENSE	On	WRITE off: sense drive status
$C0EE	R/W	WRITE	Off	Select read mode
$C0EF	R/W	WRITE	On	Select write mode
$C0EX	R/W	UDCDREG	-	UDC data register
$C0EX	W	PWMREG	-	ENABLE off and SENSE off: write PWM register
$C0EX	W	SELREG	-	ENABLE off and SENSE on: write select register
$C7XX	R/W	PORT7	On	Enable internal port 7
$CFFF	R/W	PORT7	Off	Disable internal port 7
$C800-$CBFF	R/W	-	-	PORT7 on: bank-switched RAM for internal port 7
$CC00-$CFF7	R	-	-	PORT7 on: bank-switched ROM for internal port 7
$CFF8	R	UDCDREG	-	PORT7 on: UDC data register

Table 6-1 UDC I/O Locations

Bit	Name	Function
7	SIDESEL	Set to select side 1, reset to select side 0 (for double-side drives only)
6	P7ROM2	Port 7 ROM bank select bit 2
5	P7ROM1	Port 7 ROM bank select bit 1
4	P7ROM0	Port 7 ROM bank select bit 0
3	P7RAM0	Port 7 RAM bank select bit
2	DRIVE35	Set for 3.5" drive, reset for 5.25" drive
1	RDYMODE	Set to enable ready mode
0	-	Unused

Table 6-2 UDC select register format

The disk drives which are supported by the computer can be classified into two categories: intelligent and non-intelligent.

Intelligent disk drives have dedicated microprocessor and hardware for controlling the disk read/write operation and it only receives high-level commands from the host computer. These include the Unidisk 5.25/3.5 compatible disk drives.

All the other disk drives supported by the computer are non-intelligent in the sense that all or most of the disk read/write operations have to be directly controlled by the CPU.

The following sections will describe the interfaces with the various types of disk drives in detail.

6.3 Interfacing with non-intelligent 5.25" disk drives

These refer to LASER FD-100c compatible 5.25" disk drives. They are single-sided and have a formatted storage capacity of 143K-bytes.

6.3.1 Moving the read/write head

The stepper motor used in the disk drive has four phases, namely PHASE0, PHASE1, PHASE2 and PHASE3. To position the magnetic head over a particular track accurately, you must control the voltage applied to these phases properly.

The voltage applied to the four stepper phases can be turned on or off independently by accessing (read or write) the I/O locations from $C0E0 to $C0E7. These locations are paired into four groups, with each group controlling one of the four phases as shown in Table 6-1.

To move the head outwards, i.e. away from the centre of the disk, the stepper phases must be turned on and then off one by one in descending order as shown in Fig. 6-3.

PHASE0	PHASE1	PHASE2	PHASE3	Track
Off	Off	Off	On	T
Off	Off	On	Off	T-0.5
Off	On	Off	Off	T-1
On	Off	Off	Off	T-1.5
Off	Off	Off	On	T-2
Off	Off	On	Off	T-2.5
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.

Fig. 6-3 Stepper phase sequence to move the magnetic head outwards

Reversing the phase sequence, i.e. turning the stepper phases on and then off one by one in ascending order, will move the magnetic head towards the centre of the diskette.

PHASE0	PHASE1	PHASE2	PHASE3	Track
On	Off	Off	Off	T
Off	On	Off	Off	T+0.5
Off	Off	On	Off	T+1
Off	Off	Off	On	T+1.5
On	Off	Off	Off	T+2
Off	On	Off	Off	T+2.5
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.

Fig. 6-4 Stepper phase sequence to move the magnetic head inwards

The above phase sequences will cause the shaft of the stepper motor to be rotated in steps of 90 degrees or 1/4 revolution. Each "step" of the stepper motor corresponds to a "half-track" on the diskette. To move the read/write head by one track, the stepper motor has to be "stepped" twice. Due to the inertia of the mechanical parts in the disk drive, the stepper motor will not respond immediately after new inputs are applied to the four phases. This latency is usually referred to as the "track-to-track seek time".

In addition, the magnetic head will oscillate at its new position and it requires a certain amount of time before it can settle down. This is called the "head-settling time".

As a result, the inputs to the stepper phases must be applied for a sufficiently long period of time in order to hold the magnetic head at its new position steadily. Typically, this requires about 20 ms. However, this value is different for disk drives of different manufacturers and is one of the criteria for evaluating the quality of the drive mechanisms.

The following is an example program for recalibrating the magnetic head to track 0, i.e. the outermost track.

```
;
;  Read/write head recalibration routine
;
        LDA $C0EE     ;  Select read
                         mode
        LDA $C0EA     ;  Select drive 1
        LDA $C0E9     ;  Enable disk
                         drive
        LDX #80       ;  Pull head
                         outwards for
                         80 half-tracks
```

```
LOOP1   TXA
        PHA             ;  Save track
                           count
        AND #$03        ;  Extract phase
                           number from
                           two LSB's
        ASL
        TAX
        LDA $C0E1,X     ;  Turn on
                           selected phase
      LDY #20           ;  Delay for 20 ms
;
;  Time delay loop
;  - each pass of the loop takes about 1 ms
;  - this section of code should not cross
;     page boundary
;
LOOP2   TYA             ;  2 cycles
        LDY #198        ;  2 cycles
LOOP3   DEY             ;  2 cycles
        BNE LOOP3       ;  2 cycles (+ 1
                           for successful
                           branch)
        TAY             ;  2 cycles
        DEY             ;  2 cycles
        BNE LOOP2       ;  2 cycles (+ 1
                           for successful
                           branch)
```

```
LDA $C0E0,X   ;  Turn off
                 selected phase
PLA           ;  Recover track
                 count
TAX
DEX           ;  Next
BNE LOOP1     ;  Done ?
LDA $C0E8     ;  Yes, disable
                 disk drive
RTS
```

6.3.2 Reading data from the disk drive

The UDC is configured to operate in read mode by turning the "WRITE" switch off.

In the read mode, the UDC can read two types of data from the disk drives, namely drive status and disk data, selected by the "SENSE" switch. The data is stored in the UDC data register, the contents of which can be read at locations $C0E0 to $C0EF.

If the "SENSE" switch is turned on, the drive status, which is the write-protect status in the case of LASER FD-100c compatible 5.25" drives can be read at bit 7 of the UDC data register.

If bit 7 is on, the diskette in the drive is write-protected. Otherwise, it is not. Bit 0 to bit 6 of the data register is not used when sensing drive status.

Notice that the drive status cannot be read correctly at locations $C0E8 (disable drive), $C0EC (read disk data) or $C0EF (turn on write mode).

Depending on whether drive 1 or drive 2 is currently selected, reading $C0EA (select drive 1) or $C0EB (select drive 2) may cause the wrong status to be sensed.

Reading from locations $C0E0 to $C0E7 may disturb the voltages applied to the four stepper phases and should also be avoided.

As a result, the drive status can only be sensed reliably at locations $C0E9 (enable drive), $C0ED (sense drive status) and $C0EE (select read mode).

If the "SENSE" software-switch is off, the data stored on the track below the read/write head can be read at the UDC data register.

However, since the data are recorded serially on the diskette bit by bit, how can the UDC identify the start of a valid disk data byte?

The answer is that the hardware requires each valid disk data byte to have the MSB set. As a result, you can determine whether the UDC data register contains a valid disk data by polling bit 7 of the data register. If it is off, then the data read is not valid and should be discarded. Otherwise, it is a valid disk data and can be stored somewhere or used for further processing.

However, it should be noted that the contents of the UDC data register will only be held constant for about 10 us once a valid disk data is received. As a result, the UDC data register polling loop should take less than 10 us in order to avoid loss of data.

Again, the content of the data register should not be read from locations $C0E0 to $C0E7 (stepper phases), $C0E8 (disable drive), $C0EA/$C0EB (drive select), $C0ED (sense drive status) and $C0EF (write mode).

Instead, disk data should be read from locations $C0E9 (enable drive), $C0EC (read disk data) and $C0EE (read mode). The following is an example of a 7 us polling loop for reading a byte of data from drive 1:

```
        LDA $C0EE     ; Select read mode
        LDA $C0EA     ; Select drive 1
        LDA $C0E9     ; Enable drive
          .
          .
          .
READ    LDA $C0EC     ; Read UDC data
                        register (4
                        cycles)
        BPL READ      ; Again if data
                        not valid ( 3
                        cycles)
          .
          .
          .
```

Data bytes will be transferred from the disk drive to the UDC data register every 32 us. As a result, the user has less than 32 us for processing the data read from the data register. The programmer should pay particular attention to this timing restriction in order to prevent loss of data.

6.3.3 Writing data to the diskette

The UDC enters write mode if the "WRITE" software-switch is turned on. To write a byte of data to the diskette, the user only needs to write the data to the UDC data register.

In order to prevent the states of the software-switches from being modified, the user should only write the data to the locations $C0E9 (enable drive), $C0EC, $C0ED and $C0EF (write mode).

The data in the UDC data register will be shifted out bit-by-bit to the disk drive every 4 us. This time period is called the "bit-cell time". It takes 32 us to shift out the entire 8-bit data byte written to the data register.

As a result, to write a contiguous block of data to the disk drive, it is required to load the UDC data register with new data bytes in exactly 32 us intervals. Otherwise, zero's will be written to the disk drive after all the data bits in the data register has been shifted out.

The following is an example program to write a block of 256-byte data to drive 1.

```
        LDA $C0EE     ;  Select read mode
        LDA $C0EA     ;  Select drive 1
        LDA $C0E9     ;  Enable disk
                         drive
            .
            .
            .
        LDA $C0ED     ;  Sense write-
                         protect
        BMI EXIT      ;  Skip writing if
                         disk write-
                         protected
        LDY #0        ;  Initialize buffer
                         pointer

;
;  The following section of code should be
;  assembled in the same page in memory
;
WRITE   LDA (BUFFER),Y ;  Get data to
                          be written
                          (5 cycles)
        STA $C0EF     ;  Write to data
                         register (4
                         cycles)
        PHA           ;  (3 cycles)
        PLA           ;  (4 cycles)
        PHA           ;  (3 cycles)
        PLA           ;  (4 cycles)
```

```
                NOP             ;  (2 cycles)
                NOP             ;  (2 cycles)
                INY             ;  Next entry in
                                   buffer (2 cycles)
                BNE WRITE       ;  Done ? (2 cycles,
                                   + 1 for
                                   branching)
;
EXIT            .
                .
                .
```

After enabling the disk drive, sufficient time should be allowed for the spindle motor to come up to speed before writing to the diskette. Otherwise, the data written cannot be read correctly at normal speed.

After finished with writing, you should switch the UDC back to read mode. Otherwise, zero's will be shifted out continuously and may possibly over-write some of the data stored on the diskette.

As mentioned before, the UDC hardware requires each data byte written to the disk drive to have the MSB set so that the read circuit can identify the data boundary correctly.

Besides, in order to read the data back reliably, it is also required that there should be at most two adjacent zero's in a data byte.

As a result, not all of the 256 combinations of an 8-bit byte can be used. It is necessary to scramble and encode the user data into a form which satisfies the above requirements before actually writing to the diskette. This is known as "pre-nibblization". The reverse process of transforming the raw data read from the diskette back to the original form is called "post-nibblization".

The nibblization process is done in software instead of hardware for greater flexibility. There are numerous methods of implementation and will not be discussed here.

6.4 Interfacing with non-intelligent 3.5" disk drives

These refer to LASER FD-356 compatible 3.5" disk drives They are double-side, double-density drives with a formatted storage capacity of 800 K-bytes. The control sequences of 3.5" drives are different from that of 5.25" drives. Two additional registers are required for 3.5" drive interface, namely PWM register ("PWMREG") and select register ("SELREG").

They are write-only registers which occupy the same set of addresses as the UDC data register, i.e. $C0E0 to $C0EF. However, they are write-enabled only if the "ENABLE" switch is turned off. This is to prevent their contents from being modified when writing data to the diskette through the UDC data register, during which the drive must be enabled.

When "ENABLE" is off, the data written to $C0EX will be transferred to either the PWM register or the select register depending on the state of the "SENSE" software-switch. If "SENSE" is off, data will be written to the PWM register. Otherwise, data will be written to the select register.

Notice that when data is written to either the PWM or the select register, the contents of the UDC data register will also be modified at the same time. However, this is permitted because the disk drive is disabled at that moment so that there will not be any read or write operations in progress.

6.4.1 Reading drive status

The status of the 3.5" drive can be read at bit 7 of the UDC data register as for 5.25" disk drives. There are totally 9 drive status signals which can be sensed, selected by PHASE0, PHASE1, PHASE2 and SIDESEL as shown in Table 6-3.

SIDESEL	PHASE2	PHASE1	PHASE0	Status	Description
0	0	0	0	DIR	Zero if step direction is in
0	0	0	1	STEPOK	One if finished stepping
0	0	1	0	MTRON	Zero if spindle motor is on
X	1	1	0	SIDES	One if double-side drive
X	1	1	1	DRVIN	Zero if drive is connected
1	0	0	0	DSKIN	Zero if a disk is inserted
1	0	0	1	WRTPRT	Zero if disk write-protected
1	0	1	0	TRACK0	Zero if head is at track 0
1	0	1	1	TACHO	For monitoring speed of spindle motor (60 pulses/revolution)

Table 6-3 Selecting 3.5" drive status to be sensed

To sense a particular drive status, first of all you have to enable the 3.5" disk drive. For other models in which the 3.5" disk drive is attached to the external drive connector, this can be done by turning the "ENABLE" switch and "DRIVE2" switch on.

Secondly, you should select the particular drive status to be sensed by setting the software-switches and register bit according to Table 6-3.

Finally, you should turn the "WRITE" switch off and "SENSE" switch on. The appropriate drive status can then be read in bit 7 of the UDC data register.

The status "DIR" indicates the current setting of the stepper motor direction control. If it is zero, the read/write head will move towards the centre of the disk at the occurrence of the next step pulse. Otherwise, the read/write head will move away from the disk centre at the next step pulse.

Immediately after sending a step pulse to the disk drive, you can check whether the stepping operation has finished or not by checking the "STEPOK" status. It will be set low after the application of a step pulse and will return high after stepping is finished.

If a disk is inserted in the disk drive and the spindle motor is turned on, the "MTRON" status will be read as a zero. Otherwise, it will be read as one.

The "SIDES" status indicates the number of heads in the disk drive. For single-side drives, this will be read as a zero. For double-side drives, this will be read as one.

The "DRVIN" status indicates whether the selected disk drive is actually connected or not. This should always be read as a zero if the drive is properly connected to the computer.

If the disk inserted in the drive is write-protected, the "WRTPRT" status will be read as zero. Otherwise, it will be read as one.

The "TRACK0" status will be read as zero if and only if the read/write head is at the outermost track on the disk.

The rotational speed of the spindle motor can be monitored by reading the "TACHO" status. Every revolution of the spindle motor will generate 60 pulses at this status bit. The pulses have a 50% duty cycle.

6.4.2 Sending commands to the disk drive

Unlike 5.25" disk drives, the spindle motor and stepper motor of 3.5" disk drives cannot be controlled directly. Instead, control is effected by sending special command sequences to the disk drive. There are totally six valid commands which can be send to the disk drive, selected by "SIDESEL", "PHASE2", "PHASE1" and "PHASE0" as shown in Table 6-4.

SIDESEL	PHASE2	PHASE1	PHASE0	Command	Description
0	0	0	0	DIRIN	Set step direction to in
0	1	0	0	DIROUT	Set step direction to out
0	X	0	1	STEP	Step the read/write head
0	0	1	0	MOTORON	Turn on spindle motor
0	1	1	0	MOTOROFF	Turn off spindle motor
0	1	1	1	EJECT	Eject the diskette

Table 6-4 Selecting 3.5" drive control commands

To send a particular command to the 3.5" disk drive, first of all you should enable the disk drive as described in the previous section. Then, depending on the command, you should set "SIDESEL", "PHASE2", "PHASE1" and "PHASE0" to the appropriate states according to table 6-4. The command is not transferred to the disk drive until "PHASE3" is turned on and then off again.

"DIRIN" sets the stepping direction of the read/write head to "in", i.e. towards the centre of the diskette.

"DIROUT" sets the read/write head stepping direction to "out", i.e. away from the centre of the diskette.

"STEP" moves the read/write head in the pre-defined direction (using the "DIRIN" and "DIROUT" command) by one track.

"MOTORON" and "MOTOROFF" turns the spindle motor on and off respectively.

"EJECT" causes the diskette to be ejected automatically. In order to eject the diskette successfully, "PHASE3" should be turned on for at least 750 ms.

6.4.3 Reading disk data

To read the data stored on the diskette, first of all you should enable the disk drive as described before. Secondly, you should configure the UDC to read mode by turning "WRITE" and "SENSE" off. Thirdly, you should inform the disk drive that disk data instead of drive status is to be read by turning off "PHASE0" and "PHASE1" and turning on "PHASE2". Finally, you should select one of the two disk surfaces to be read by properly setting the "SIDESEL" bit of the UDC select register. The disk data can then be read at the UDC data register as in 5.25" disk drive.

However, since the recording density in 3.5" drive is twice that of 5.25" disk drive, the bit cell time is reduced from 4 us to 2 us. As a result, valid disk data will be shifted into the UDC data register every 16 us.

As described before, determining whether the contents of the UDC data register is a valid disk byte can be done by a program loop which polls bit 7 of the data register for a one. This is perfectly alright for 5.25" disk drives in which there are plenty of processing time available (32 us) between arrivals of valid disk data.

However, for 3.5" disk drive, things are not that simple. Since the program loop for reading disk data usually includes instructions for transforming and storing the disk data, updating the loop counter etc., it may not be possible to fit in a 16 us time interval if polling is used.

As a result, a special mechanism for reading disk data from 3.5" disk drive is employed in the UDC which is known as the "READY" mode. It is entered by setting the "RDYMODE" bit of the UDC select register to one.

In the "READY" mode, whenever the CPU reads the UDC data register through the location $CFF8 (after accessing any location from $C700 to $C7FF), the read cycle will be extended automatically if bit 7 of the UDC data register is zero. The read cycle can only be completed after bit 7 of the UDC data register goes high, indicating a valid disk data is available.

With this feature, you don't have to poll the UDC data register for a valid disk data and hence the processing time wasted on this task previously can now be spent on other operations. This makes a 16 us program loop for reading disk data feasible.

6.4.4 Writing data to the disk drive

The procedure of writing data to a 3.5" disk drive is similar to that of 5.25" disk drive except that the data written to the UDC data register is shifted out every 2 us.

To write data to the disk drive, first of all you should enable the drive by setting "DRIVE35" to one, turning on "ENABLE" and setting "DRIVE2" to the appropriate state depending on whether you have a built-in or external 3.5" disk drive. Secondly, you should turn on the spindle motor as described previously. Thirdly, you should configure the UDC to write mode by turning on the "WRITE" software-switch. You should then proceed to store the stream of data to be written to the diskette into the UDC data register in exact 16 us intervals.

There is another difference between 5.25" and 3.5" disk drives in regards to drive speed and recording density. For 5.25" disk drive, the rotational speed of the spindle motor is fixed at about 300 rpm (revolution per minute). Since for constant angular velocity, the linear velocity of a particular point on the disk surface is directly proportional to its distance from the disk centre, it follows that the "width" of a bit cell on a particular track is proportional to the radius of that track. As a result, the recording density is not uniform on the entire diskette (higher on inner tracks) so that some storage capacity (on the outer tracks) is wasted.

For 3.5" disk drives, the rotational speed of the spindle motor is variable from 300 rpm to 600 rpm. The motor speed is software-controllable through the PWM register of the UDC. The larger the value written to the PWM register, the higher the rotational speed of the spindle motor. The actual speed can be figured out by reading the "TACHO" status as described in section 6.4.1.

With this speed control capability, it is possible to control the spindle motor in the way such that it rotates at lower speed in outer tracks and higher speed in inner tracks while keeping the linear velocity of the disk surface relative to the read/write head constant. As a result, the recording density is the same over the entire diskette and the maximum storage capacity is exploited.

6.5 Interfacing with intelligent disk drives

Intelligent disk drives include the Unidisk™ 5.25 and Unidisk™ 3.5 compatible disk drives. They have built-in microprocessor, ROM, RAM and other circuitries for controlling the disk read/write operations and are effectively microcomputers on their own. They only receive and process high-level commands (such as read status, format, read block and write block etc.) from the host computer.

The intelligent disk drives communicate with the host computer through the signal lines on the external drive connector. Control parameters and data are exchanged between the computer and the disk drive in the form of data packets which are transmitted and received byte-by-byte in a specially encoded format through the UDC data register. These data bytes have similar format as the disk data, e.g. they must have their bit 7 on. Moreover, successive data bytes must be written in exact time intervals as in normal disk write operation.

The detail implementation will not be described here. For more information, please refer to the technical documentations published by the drive manufacturers.

CHAPTER 7
EXPANSION RAM

7. EXPANSION RAM

Besides the 128 K-byte system RAM that comes with the computer, it also has all the necessary hardware and firmware built-in for accommodating up to 1 M-byte expansion RAM.

For the "LASER 128 EX", a fully-socketed expansion RAM card is built-in for supporting up to 32 dynamic RAM chips.

For the "LASER 128", the expansion RAM card is purchased separately. For more information on installation of the RAM board, please refer to the user's manual that comes with the RAM card.

7.1 Accessing the expansion RAM

Unlike the system RAM, the expansion RAM is not directly addressable. Instead, it is "indirectly" addressed through a 20-bit read/write address register mapped to I/O locations $C0D0, $C0D1 and $C0D2 of the address space of 65C02 as shown in Table 7-1.

Address	Operation	Register	Description
$C0D0	R/W	XAREGL	Low-byte of address register
$C0D1	R/W	XAREGM	Mid-byte of address register
$C0D2	R/W	XAREGH	Hi-nibble of address register in bit 0 to bit 3; bit 4 to 7 unused and read as 1's
$C0D3	R/W	XDREG	Expansion RAM data register
$C5XX	R/W	XRAME	Expansion RAM enable; data written to the register is not used

Table 7-1 Expansion RAM control I/O locations

To address a particular expansion RAM location, you only need to store the 20-bit address into the expansion RAM address register.

If the expansion RAM is enabled, the addressed expansion RAM location can be read from or written to through the data register at $C0D3.

Moreover, the content of the 20-bit address register will be incremented automatically each time the data register is accessed for either read or write. This feature can reduce the software overhead spent on moving a large block of data to or from contiguous expansion RAM locations and thus speeding up the data transfer operation.

Immediately after power-up or "CTRL-RESET" is pressed, the expansion RAM is disabled. The address register can still be read or write correctly. However, the function of the data register is inhibited. Data cannot be transferred to or from the addressed expansion RAM location through the data register and the address register will not be incremented after the data register is accessed.

To enable the expansion RAM, you only need to access any location from $C500 to $C5FF, where part of the firmware driver for the expansion RAM is stored.

The entire expansion RAM interface can be disabled by throwing the "INT/EXT PORT 5" switch located inside the ROM door on the bottom cabinet to the "EXT PORT 5" position. In this case, both the address register and data register will lost their functions. Moreover, the firmware driver for the expansion RAM will also be disabled.

7.2 Applications of the expansion RAM

As described before, the expansion RAM is not directly addressable by the CPU. As a result, it cannot be used as program memory for executing programs. Instead, it is specially designed for efficient storage and retrieval of a large block of data which is accessed sequentially.

The most common application of the expansion RAM is to use it as a "RAM disk", i.e. emulating a very high-speed disk system for storage of program or data files.

Before you begin to work with the programs and data stored on a diskette, you may load some or even all of the data files stored on the diskette into the expansion RAM. You can then work with this "RAM disk" instead of the actual diskette. Since no mechanical movement is involved in the operation of the "RAM disk", the information can be accessed much faster than a real disk system. Moreover, program and data files from more than one diskette can be loaded into the "RAM disk" at the same time (if the size of the expansion RAM is large enough) so that no disk swapping is required.

After you have finished with your work, you can then copy the contents of the "RAM disk" back to the actual diskette. Notice that since the diskette is only accessed twice in the entire working section, damage to the disk surface due to mechanical movement is greatly reduced and the working life of the diskette is lengthened.

The expansion RAM is already recognized by some operating systems (e.g. Pascal version 1.3 or later versions, DOS 3.3 and ProDOS etc.) and will be formatted as a storage device automatically. However, you may also write your own device driver for the expansion RAM so as to support other application programs.

CHAPTER 8
PARALLEL PRINTER PORT AND SERIAL PORTS

8. PARALLEL PRINTER PORT AND SERIAL PORTS

A parallel printer can be connected to the DB-15 parallel printer connector on the back panel of the computer for printing hard-copies of your work.

Besides, there are also two serial interface connectors for connecting serial printers or other serial communication devices such as modems.

8.1 Parallel printer port

The computer can support both parallel printer and serial printer. The option is selected by a small slide switch on the front panel. The parallel printer port is the D-type connector next to the game port.

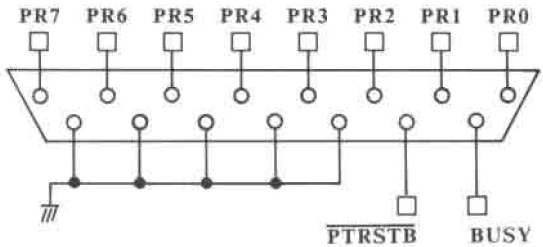


Fig. 8-1 Parallel printer port pin assignment

The timing diagram of sending data to the printer is shown in Fig. 8-2. Before sending data to the printer, the printer busy signal is checked first. If the printer is busy, the computer will wait until the printer is not busy.

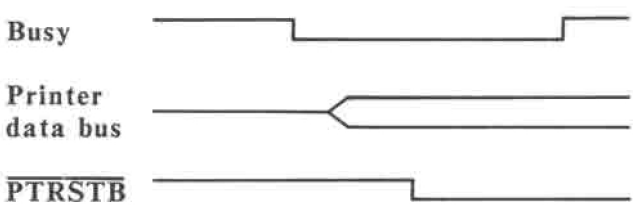


Fig. 8-2 Parallel printer control timing

The hardware page locations for manipulating the parallel printer are listed in Table 8-1.

Address	Operation	Description
$C09X	W	Send PRINTER STROBE to printer
$C1C1	R7	Bit 7 = 1; printer Busy

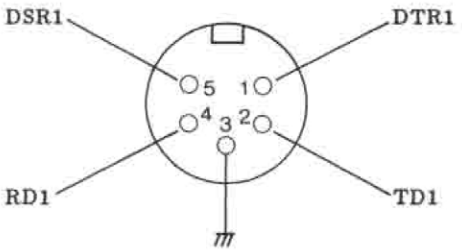
Table 8-1 Hardware page locations of parallel printer

8.2 Serial ports

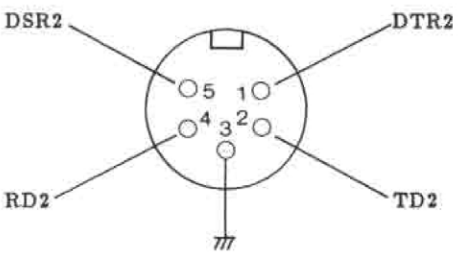
The computer has two serial ports built-in. Port 1 is used for the serial printer and port 2 is for the modem. However, with the help of system utility programs, the characteristics of the two I/O ports can be reconfigured. The hardware page locations of the two serial I/O ports are listed in Table 8-2. The pin assignments of the serial interface connector are shown in Fig. 8-3.

Address	Description
$C098	Port 1 ACIA receive/transmit data register
$C099	Port 1 ACIA status register
$C09A	Port 1 ACIA command register
$C09B	Port 1 ACIA control register
$C0A8	Port 2 ACIA receive/transmit data register.
$C0A9	Port 2 ACIA status register
$C0AA	Port 2 ACIA command register
$C0AB	Port 2 ACIA control register

Table 8-2 Hardware page locations of the serial I/O ports



Serial Port 1



Serial Port 2

Pin	Pin name	Description
1	DTR1 DTR2	Data Terminal Ready output
2	TD1/ TD2	Transmit Data output
3	GND	Power and signal common ground
4	RD1/ RD2	Receive Data input
5	DSR1/ DSR2	Data Set Ready input

Fig 8-3 Pin assignment of serial port 1 and 2.

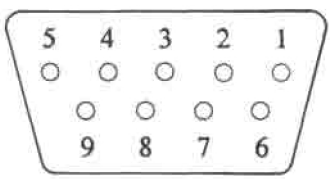
CHAPTER 9
GAME PORT

9. GAME PORT

The game control port can be functionally divided into 2 parts, namely:

a) The switch and analog (paddle) inputs.

b) The mouse inputs.



1 MOUSE SIGNATURE / GAMESW1
2 + 5V
3 GND
4 XDIR
5 XINT / PDL0
6 N.C.
7 MOUSE BUTTON / GAMESW0
8 YDIR / PDL1
9 YINT

Fig. 9-1 Game Port Connector Pin assignment

9.1 Switch and analog inputs

The connection of the paddle and switch input is shown in Fig. 9-2.

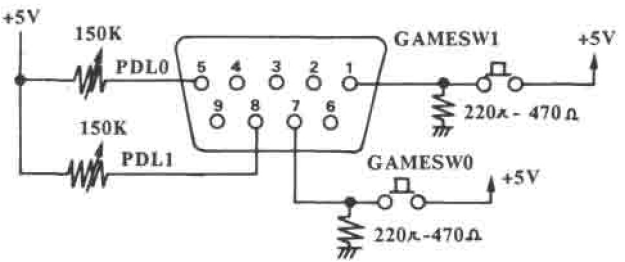


Fig. 9-2 Paddle and switch input requirement

The hardware page locations are listed in Table 9-1.

Address	Operation	Description
$C061	R7	Bit 7=1 and Bit 7 of $C063 equals 1, GAMESW0 is pressed; if only bit 7=1 then ▵ is pressed
$C062	R7	Bit 7=1; ▲ or GAMESW1 is pressed
$C063	R7	Bit 7=1 and bit 7 of $C061 is 1, then GAMESW0 is pressed. If bit 7=0 then mouse button is pressed.
$C064	R7	Analog (paddle 0) input

Address	Operation	Description
$C065	R7	Analog (paddle 1) input
$C07X	R/W	Trigger paddle timer

Table 9-1 Hardware page locations of switch and analog input

The analog inputs can be connected to two 150K paddles as game control. Usually, a two-axis joystick is connected to these inputs. The keys ▲ and △ , connect to these two locations permanently. The switch inputs GAMESW0 and GAMESW1 also connect to these locations. However, to distinguish these game switch inputs from mouse button, $C063 has to be read. When reading $C063 gives a one in bit 7, GAMESW0 is pressed. If the mouse button is pressed, the value will be 0 for bit 7 of $C063.

9.2 Mouse input

The game port of the computer also accepts mouse as its input. The connection of the mouse to the game port is shown in Fig. 9-3.

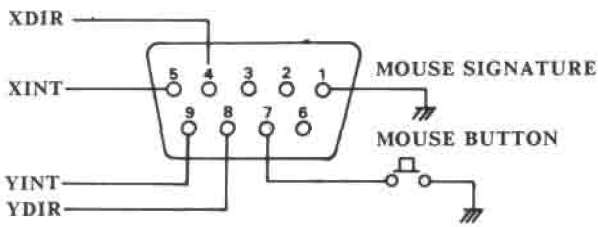
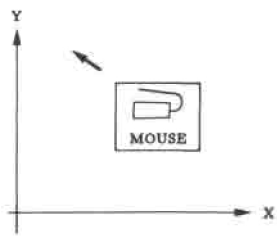


Fig. 9-3 Connection of game port to mouse

When a mouse is moved along a flat surface, square pulses are output on pin XINT, XDIR, YINT and YDIR. The XINT and YINT are the interrupt signals to the MPU. With software control, either the rising or the falling edge of the mouse interrupt signal can cause interrupt to the MPU. The direction of movement of the mouse can be observed from the XDIR or YDIR signals.

To illustrate the operation, we can study an example. The XINT rising-edge causes the interrupt. When the MPU receives the interrupt, it immediately checks the XDIR. If the XDIR is at a high level, the mouse moves in -X direction. Similarly, if the value of YDIR is '0' immediately after YINT interrupt edge, the mouse must move in a +Y direction.



Mouse movement direction

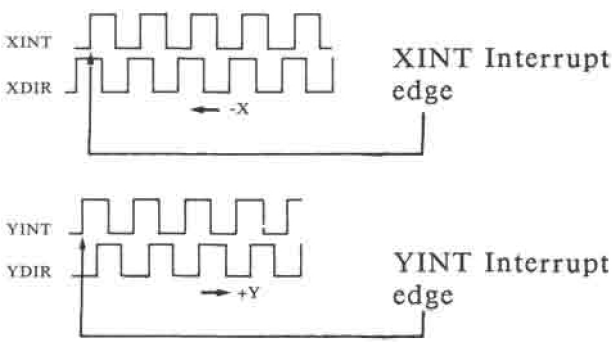


Fig. 9-4 Example to illustrate the mouse operation

The mouse signature pin is connected to ground so as to disable the paddle analog input. The mouse button status can be read from $C063.

The vertical blanking signal (VBLK) can cause an interrupt to the MPU if this interrupt source is enabled. The programmer can then enable the mouse and update the mouse position information if there is any mouse movement. The hardware page locations which control all these interrupts are listed in Table 9-2.

Address	Operation	Function
$C063	R7	Bit 7=0: mouse button pressed
$C0C0	W	Select rising-edge of XINT as int. source
$C0C1	W	Select falling-edge of XINT as int. source
$C0C2	W	Select rising-edge of YINT as int. source

Address	Operation	Function
$C0C3	W	Select falling-edge of YINT as int. source
$C0C4	W	Disable XINT and YINT interrupt
$C0C5	W	Enable XINT and YINT interrupt
$C0C6	W	Disable VBLK interrupt
$C0C7	W	Enable VBLK interrupt
$C0C8	R7	Read selected XINT int edge; Bit 7=1: falling-edge
$C0C9	R7	Read selected YINT int. edge; Bit 7=1: falling-edge

Address	Operation	Function
$C0CA	R7	Read XINT and YINT int. enable flag Bit 7=1; XINT and YINT enabled
$C0CB	R7	Read VBLK int. enable flag; Bit 7=1: VBLK int. enabled
$C0CC	R7	Read XINT interrupt status; Bit 7=1: XINT interrupt occurred
$C0CD	R7	Read YINT interrupt status; Bit 7=1: YINT interrupt occurred

Address	Operation	Function
$C0CE	R7	Read VBLK interrupt status; Bit 7=1: VBLK interrupt occurred
$C0CF	W	Reset XINT and YINT interrupt status
$C07X	W	Reset VBLK interrupt status

Table 9-5 Mouse Hardware Page Locations

There are routines in the Monitor program to handle the mouse movement. The programmer may use these routines instead of writing their own.

CHAPTER 10
SYSTEM FIRMWARE

10. SYSTEM FIRMWARE

The firmware that is stored permanently in the ROM of the computer can be divided into three main sections:

1.	$C100 - $CFFF	I/O device drivers
2.	$D000 - $F7FF	BASIC interpreter
3.	$F800 - $FFFF	System kernel

10.1 System Kernel

Most of the "System Kernel" resides at $F800 - $FFFF, though a few system routines lie in other areas of ROM. The System Kernel forms the low-level basis for how the computer works. It can be functionally divided into three main categories:

1. Power-up, interrupt, BRK, and CONTROL-RESET handling
2. Miscellaneous routines available for use by application programs
3. The System Monitor program

10.1.1 Power-up and CONTROL-RESET

When the computer is turned on, circuitry inside the computer notices that the computer is being turned on and responds by asserting the RESET line for a moment (This action is equivalent to a user pressing CONTROL-RESET).

Any time RESET is asserted, whether by turning the computer on or by pressing CONTROL-RESET, other circuitry inside the computer responds instantly with several actions (described in detail elsewhere in this manual). The most important actions are that the main bank of RAM is enabled, and the ROM area is enabled for reading. This puts the computer in a "standard" memory configuration after every Reset.

The 65C02 microprocessor also responds to the Reset. It abandons whatever it was previously doing, and reads the two-byte address stored in locations $FFFC,$FFFD. This two-byte address comes from the ROM, since the ROM is always enabled at this point.

The microprocessor begins executing the code found at the address it just read. This code is also in the ROM area.

The Reset-handling code in ROM begins by initializing several system locations in memory, and setting up a standard 40-column text-only video display. It checks the setting of the 40/80 column switch to determine whether or not to "map in" the 80-column firmware at $C300-$C3FF to make the 80-column display available.

It then checks if the hollow-triangle key is currently being held down (part of CONTROL-HOLLOW TRIANGLE-RESET). If it is, then it "blasts" one byte of memory in every page from page $01 to page $BF. The byte blasted in page $03 is the "application Reset checksum" (described below). The stack pointer is also reset to $FF.

Next, the Reset handler beeps the speaker for about a tenth of a second. It then checks if the "P" key (upper-case or lower-case P or CTRL-P) is being held down (CONTROL-P-RESET). If it is, then it goes to execute the port configuration program described in the user's manual

Otherwise, it next checks the "application Reset vector and checksum". An application can store the address of its own Reset handler into the application Reset vector at locations $3F2,$3F3. For the vector to be recognized, the application must also exclusive-OR the high-byte of the address with the constant $A5, and store the result into the application Reset checksum at location $3F4.

The Reset handler itself exclusive-ORs the high-byte of the application Reset vector with $A5 and compares it with the application Reset checksum. If they are identical, then the Reset handler "knows" that the Reset vector was correctly set up. If it was, the Reset handler does one more check: if the Reset vector points to $E000, then it changes the vector to point to $E003 and jumps directly to $E000 (This allows BASIC to be "coldstarted" on the first Reset or power-up, and "warmstarted" on subsequent Resets). If the vector does not point to $E000, then the Reset handler simply jumps to the address contained in the vector.

Note: The application Reset vector will not match the checksum if the computer was just turned on (since nothing has initialized it yet) or if CONTROL-HOLLOW TRIANGLE-RESET was pressed (since the byte-blasting that occurred stored an invalid Reset checksum).

If the application Reset vector did not match the checksum, then the Reset handler takes several more steps. It initializes the application Reset vector and checksum to correctly point to $E000 (so that if CONTROL-RESET is subsequently pressed, then BASIC will be coldstarted). It clears the screen and displays the ROM version number and a copyright message at the top of the screen. It then checks if a bootable disk device is present in port 7 (either internal or external depending on the setting of the "INT/EXT PORT 7" switch). If one is, then the handler jumps to it so that the disk device will boot. If not, then the port 6 floppy disk driver is called to boot a disk in the built-in floppy disk drive.

10.1.2 Interrupt and BRK Handling

Interrupt handling can usually be ignored by application programs on the computer. The mouse and the two serial ports on the computer can generate interrupts, but these are normally handled transparently by the system firmware.

When the 65C02 microprocessor has determined that an IRQ interrupt has occurred or a BRK instruction has been executed, it pushes the program counter and status register values onto the stack, reads the two-byte address from locations $FFFE,$FFFF, and jumps to that address. Note, however, that locations $FFFE,$FFFF can be occupied by either ROM, main RAM, or auxiliary RAM. If an application program is using either the main or auxiliary upper 16K of RAM and requires interrupts to work and/or BRK instructions to be handled correctly, it must store a valid interrupt handler address into $FFFE, $FFFF of these areas. (Initializing the mouse will automatically copy the address of the built-in interrupt handler into both the main and auxiliary upper 16K RAM areas).

In the built-in firmware in ROM, $FFFE,$FFFF point to the computer interrupt handler that begins in the $C4 page. In the computer, $C400-$C4FF is shared between mouse routines and interrupt handler routines. This page of ROM is always mapped in and always available, regardless of what memory configuration soft-switches have been accessed. This makes it the ideal entry point for the interrupt handler, because it means that interrupts can occur and be handled correctly with any memory configuration.

The interrupt handler in page $C4 saves the contents of the registers along with a byte representing the current memory configuration, then switches to main memory, ROM active. It then checks whether a BRK or a real interrupt has occurred.

If a BRK occurred, it transfers information about the BRK into several zero page locations:

$3A, $3B — Program counter value (The number stored here will actually be 2 greater than the address of the BRK that occurred.)

$44 — Memory configuration - Each bit is a boolean value (1=TRUE, 0 = FALSE)

- Bit 7 = Auxiliary zero page, stack, & upper 16K are switched in
- Bit 6 = Both 80-column store and "page 2" access are switched in
- Bit 5 = Auxiliary 48K RAM is switched in for reading
- Bit 4 = Auxiliary 48K RAM is switched in for writing
- Bit 3 = Upper 16K is switched in for reading
- Bit 2 = Bank 1 of the upper 16K is switched in

Bit 1 = Bank 2 of the upper 16K is switched in
Bit 0 = Internal $CXXX ROM is switched in

$45 Accumulator value
$46 X-register value
$47 Y-register value
$48 Processor status register value
$49 Stack pointer value

The BRK handler then reads the two-byte address from $3F0,$3F1, and jumps to the routine at this address. Unless altered by an application program, this vector will point to the "second half" of the BRK handler. This second half switches to a standard 40-column text-only display, and prints:

BRK:$nnnn

where nnnn is 2 greater than the address of the BRK instruction, then enters the System Monitor. A technically inclined user can then examine location $44-$49 to determine the circumstances surrounding the BRK.

(Note: The BRK handler will always save the correct register values into zero page, even if the BRK occurs while the auxiliary memory stack and zero page are active, and the stack needs to be switched.)

If an interrupt occurred (rather than a BRK instruction), the interrupt handler determines if the source of the interrupt was either an "invisible" mouse interrupt or a serial port handshake-line interrupt. In either case, the interrupt is simply handled internally, then control is returned back to the interrupted program.

If the source of the interrupt was either a "visible" mouse interrupt, a modem port receiver or transmitter interrupt, or an interrupt generated by a circuit card plugged into the expansion box then it pushes some return information onto the stack, reads the two-byte address stored at locations $3FE,$3FF, and jumps to this address. This vector must be initialized by the application ahead of time! Otherwise, the computer will hang or behave erratically.

The application interrupt routine should exit with an RTI instruction. The RTI will return control to the computer firmware, which will unstack the configuration information, restore the memory configuration and register values, then return back to the interrupted program.

Note: Applications which use the mouse should ALWAYS use the built-in interrupt handling firmware, rather than handling the interrupts directly themselves. The internal handling of the mouse on the computer is not identical to other computers. (A couple of current mouse-based software programs do bypass the built-in interrupt handler. The computer's mouse firmware is designed to accommodate these programs, but these accommodations should not be relied upon for future products.) Application programs which need visible mouse interrupts should connect their interrupt routines through the application interrupt vector at $3FE,$3FF rather than intercepting $FFFE,$FFFF.

10.1.3 Miscellaneous Routines

The System Kernel contains a large number of useful routines and entry points. Application programs can call these routines to help accomplish a variety of tasks. Listed below are entries for each of the supported routines. Each entry includes the routine address, a brief description of the routine, and any other information that might be necessary to use it.

WARNING: Do not attempt to call or make use of any system code that is not documented below. Any undocumented routines are subject to possible changes in future versions of the computer ROM, and will not correspond to routines in other 65C02 based computers. (A few entry points which are not documented are provided for the sake of compatibility with current software products, but these entry points are not guaranteed in future ROM versions, and should not be relied upon for future software development.) Also, do not expect 65C02 registers to contain specific values on return, unless specified below.

$C305 Get a character from the keyboard, displaying a solid box cursor using 80-column firmware routines. Returns: Acc = character typed. The 80-column firmware must be initialized for this to work. A preferred method is to initialize the 80-column firmware and obtain characters through the routine at $FD0C.

$C307 Print a character to the text display, using 80-column firmware routines. Pass in: Acc = character to print. Returns: Acc, Xreg, Yreg unchanged. The 80-column firmware must be initialized for this to work. A preferred method is to initialize the 80-column firmware and print characters through the routine at $FDED.

$C311 Move a block of data from main 48K RAM to auxiliary 48K RAM, or vice versa. Pass in: $3C,$3D contain two-byte address of first byte in block to move, $3E,$3F contain two-byte address of last byte in block to move, $42,$43 contain two-byte address of where to move block to, carry flag clear to move from auxiliary memory to main memory. The computer 40/80-column switch must be set to 80 for this routine to be available.

$C314 Switch program execution from main 48K RAM to auxiliary 48K RAM, or vice versa, also specifying which zero page and stack area to use. Pass in: $3ED,$3EE contain two-byte address of code to execute next, carry flag clear for execution in main 48K RAM, else carry flag set for execution in auxiliary 48K RAM, overflow flag clear for main zero page and stack, else overflow flag set for auxiliary zero page and stack. The 40/80-column switch must be set to 80 for this routine to be available.

$F800 Plot a block in current color on the low-resolution graphics screen. Pass in: Yreg = horizontal coordinate, Acc = vertical coordinate.

$F819 Draw a horizontal bar in current color on the low-resolution graphics screen. Pass in: Yreg = left coordinate, $2C = right coordinate, Acc = vertical coordinate.

$F828 Draw a vertical bar in current color on the low-resolution graphics screen. Pass in: Yreg = horizontal coordinate, Acc = top coordinate, $2D = bottom coordinate.

$F832 Clear the entire low-resolution graphics screen to black.

$F836 Clear the top 40 lines of the low-resolution graphics screen to black. (If in mixed text/graphics mode, this leaves the bottom four lines of text unaffected.)

$F864 Set the current color for low-resolution graphics. Pass in: Acc = color number (0-15). The color for each color number can be found in the BASIC manual.

$F871 Determine the color at a given coordinate on the low-resolution graphics screen. Pass in: Yreg = horizontal coordinate, Acc = vertical coordinate. Returns: Acc = color number.

$F941 Print a 4-digit hex number. Pass in: Acc = high byte of number, Xreg = low byte of number.

$F948	Print three spaces.
$F94A	Print 1 to 256 spaces. Pass in: Xreg = number of spaces to print.
$FB1E	Read a game paddle or one axis of a joystick. Pass in: Xreg = paddle number (0 or 1). Returns: Yreg = value of paddle (0 to 255).
$FB2F	Switch in the standard text display and initialize the window to the full screen. If 80-columns are already active, will initialize to 80-column text display, otherwise will initialize to 40-column display.
$FB39	Alternate entry to $FB2F, does not enforce display page 1 or (invisible) low-resolution graphics mode.
$FB40	Switch in the mixed low-resolution graphics and text display, and set a 4-line text window at the bottom of the screen.
$FB5B	Set the vertical cursor position. Pass in: Acc = new vertical position.

$FB6F Set the correct application Reset checksum. Pass in: $3F2,$3F3 contain desired application Reset vector address. Returns: $3F4 contains correct checksum for the Reset vector address.

$FBDD Pause for about one-hundredth of a second, then beep the speaker for about a tenth of a second. (If you want to beep the speaker repeatedly - admittedly an annoying idea - the pause is just long enough to separate the sound into distinct beeps.)

$FBC1 Set the left margin "base address" for a given line on the screen. Pass in: Acc = vertical position.

Returns: $28, $29 = address of the leftmost character position on that line. (For normal text display, $FB5F or $FC22 are the preferred entry points.)

$FBE2 Beep the speaker for about a tenth of a second (no pause).

$FBF4 Advance the cursor one position.

$FC10 Back up the cursor one position.

$FC1A Move the cursor up one line.

$FC22	Set the vertical cursor position. Pass in: $25 contains new vertical position.
$FC42	Clear from the current cursor position to the bottom of the text window.
$FC58	Clear the entire text window and place the cursor at the upper left corner of the text window.
$FC62	Move the cursor to the leftmost position on the next line down.
$FC66	Move the cursor down one line.
$FC70	Scroll the text window up one line.
$FC9C	Clear from the current cursor position to the end of the line.
$FCA8	Pause for a moment. Pass in: Acc = how long to pause. Approximate example times: $01 = 32 microseconds, $04 = 110 microseconds, $10 = 1 millisecond, $40 = 11 milliseconds, $80 = 45 milliseconds, $00 = 170 milliseconds (.17 seconds).
$FD0C	Call the current character input routine. Returns: Acc = character input.

$FD1B Character input routine for 40-column display. Blink a checkerboard cursor at the current cursor position, waiting for a key to pressed, generating a random number in locations $4E,$4F. Returns: Acc = key pressed. A preferred method is to Initialize 40-column I/O, then call $FD0C instead.)

$FD35 Character input routine, handling ESCape codes for either 40 or 80 column display.

$FD67 Print carriage return, prompt character, and read a line of text into page $02 from the current character input routine, terminated with carriage return. Handles ESCape codes, left and right arrow keys (for backspace and retype), and CTRL-X to cancel line. Returns: Xreg = number of characters in line, not including carriage return.

$FD6A Similar to $FD67, but does not print initial carriage return.

$FD6F Similar to $FD6A, but does not print prompt character.

$FD8B Clear from current cursor position to end of line, then print carriage return.

$FD8E Print carriage return to current character output port.

$FDDA Print a two-digit hex number. Pass in: Acc = number to print.

$FDE3 Print a single-digit hex number. Pass in: Acc = number to print.

$FDED Print character through current character output port. Pass in: Acc = character to print.

$FDF0 Print character to text display. Pass in: Acc = character to print. Returns: Acc, Xreg, Yreg unchanged. A preferred method is to initialize the 40-column display and print characters through the routine at $FDED.

$FE1F "Computer compatibility routine" - a "null" routine in the computer.

$FE2C Copy a block of data from one area of memory to another. Pass in: $3C,$3D contain address of first byte of block to copy; $3E,$3F contain address of last byte of block to copy; $42,$43 contain address of where to copy block to.

$FE80 Set normal (white-on-black) character display.

$FE84 Set inverse (black-on-white) character display.

$FE89 Set character input to standard 40-column routines.

$FE8B Set character input to given port. Pass in: Acc = desired port number.

$FE93 Set character output to standard 40-column routines.

$FE95 Set character output to given port. Pass in: Acc = desired port number.

$FF2D Print "ERR" and beep the speaker.

$FF3A Print CTRL-G (bell character) to the current character output port.

$FF3F	Retrieve register contents from zero page: $45 -> Acc, $46 -> Xreg, $47 -> Yreg, $48 -> Processor status register.
$FF4A	Save register contents into zero page: Acc -> $45, Xreg -> $46, Yreg -> $47, Processor status register -> $48, stack pointer -> $49.
$FF58	Documented location of an RTS instruction.
$FF59	Initialize standard 40-column display and character input/output routines, clear decimal mode, beep the speaker, and enter System Monitor program.
$FF65	Clear decimal mode, beep the speaker, and enter System Monitor program.
$FF69	Enter system Monitor program.

$FFA7	Attempt to parse a hex number from the line of text in page $02. Pass in: Yreg = offset into line of first character to parse. Returns: Acc = first non-hex-digit character; Xreg = nonzero if number found, zero if number not found; Yreg = offset into line 1 past first non-hex-digit character; $3E, $3F contain parsed number, or zero if no number to parse.
$FFFA, $FFFB	Contains address of NMI handler: $03FB. (Handler must be set up by application.)
$FFFC, $FFFD	Contains address of Reset handler in ROM.
$FFFE, $FFFF	Contains address of IRQ interrupt handler in ROM.

10.1.4 The System Monitor

Included in the System Kernel is a "System monitor", a program designed to allow technically inclined users to examine and change memory locations directly. The System Monitor includes commands for displaying the contents of memory as hexadecimal numbers, or optionally as ASCII characters and disassembled 65C02 instructions. Commands are also available for moving blocks of data from one area of memory to another, executing a 65C02 routine, and reading the 6-digit version code of the ROM. The System Monitor works strictly with hexadecimal numbers.

Warning: Because of the way input/output is mapped in the computer, changing or even examining certain areas of memory can cause the computer to hang or behave erratically. In general, you should avoid accessing locations from $C000 to $CFFF. If you want to examine these areas, you should understand the I/O mapping in this area and take appropriate care. (The I/O mapping is explained in chapter 3 of this manual.)

The three entry points into the System Monitor, as described in the last section, are $FF59,$FF65, and $FF69. The easiest way to enter the Monitor is to turn on the computer, immediately press CTRL-RESET to stop the drive, then type:

]CALL -151

(Decimal -151 is equivalent to $FF69.)

You'll see an asterisk and a blinking checkerboard cursor. You're now in the System Monitor program. Here are the System Monitor commands:

addr RETURN

To examine the content of a single memory location, type the hex address and press RETURN. You'll see the address, a dash, and the value at that address.

addr1.addr2 RETURN

To see the contents of a range of locations, type the starting address, a period, the ending address, and press RETURN. The monitor will print one or more lines, with an address, a dash, and up to eight data values on each line, covering the range you specified.

RETURN

To see the contents of a few memory locations beyond where you last looked, just press RETURN. You'll see up to eight more data values.

addr:data data data ... data RETURN

To change the contents of one or more consecutive memory locations, type the starting address and a colon, followed by one-digit or two-digit data values separated by spaces. Press RETURN after the last data value. The contents of the memory locations will be changed to the values you specified.

addrL RETURN
L RETURN

To disassemble the instructions in an area of memory, type the starting address and the letter "L", and press RETURN. Twenty lines will be printed, each with the address of the instruction, the bytes shown as ASCII characters, the bytes shown as hexadecimal numbers, and the disassembled instruction. To continue disassembling, type "L" and RETURN. (Note: because of the way various ROM areas are automatically switched in and out by the computer firmware, not all areas of ROM can be directly viewed or disassembled.)

addr1<addr2.addr3M RETURN

To move a block of data from one area of memory to another, type the starting destination address, a less-than sign, the starting source address, a period, the ending source address, and the letter "M". The data will be moved from the source area to the destination area.

addrG RETURN

To execute a 65C02 routine, type the address of the routine and the letter "G". The routine will execute. If it ends with an RTS, the routine will return to the System Monitor.

CTRL-V RETURN

Displays the six-digit ROM version code.

Multiple commands can also be entered on one line, separated by spaces. If you want to enter bytes into memory and do another command on the same line, you can separate the bytes entered from the other commands with the letter "N". For example:

*addr:data data data N addrG RETURN

10.2 INPUT/OUTPUT ROUTINES

The input/output routines in the computer are divided into 8 ports, numbered 0 to 7. They are:

Port 0 - built-in 40-column display
Port 1 - printer port
Port 2 - serial communications (modem) port
Port 3 - built-in 80-column display
Port 4 - mouse port
Port 5 - 1 M expansion RAM
Port 6 - 5.25" disk drive
Port 7 - 3.5" disk drive

(Port 5 and port 7 firmware may be inhibited by the "INT/EXT PORT 5" and "INT/EXT PORT 7" switch.)

Most of the I/O firmware resides in the area $C100-$CFFF.
Below is a memory map for this area.

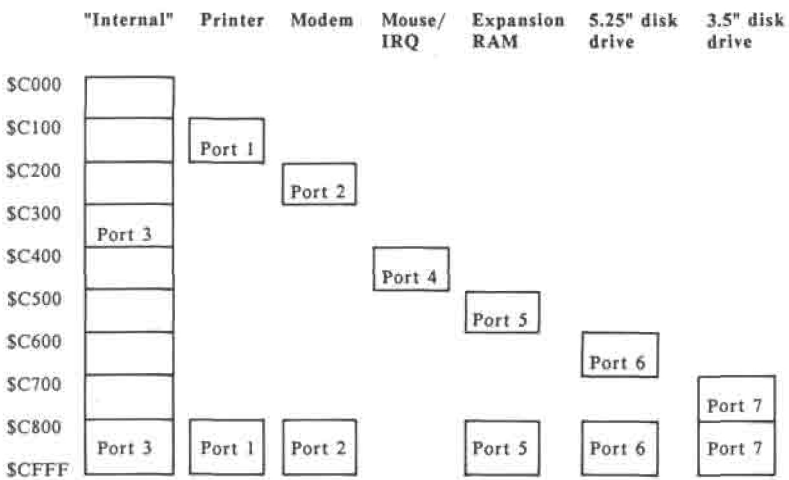


Fig. 10-1 Logical I/O address mapping

The computer hardware allows different portions of ROM to be mapped in and out. The actual switching arrangement is described in chapter 3 of this manual, but is really unimportant as far as most application programs are concerned. The ROM is divided into an "internal" portion that spans $C100 to $CFFF, and several 256-byte areas for port 1 to 7. Except for port 4, all the other ports can also switch in additional areas (only one at a time!) at $C800 to $CFFF.

The "internal" portion of ROM contains additional internal code for 40 and 80 column displays, mouse & IRQ routines, etc. This portion is switched in only by the firmware as long as necessary, then switched out again to make the I/O ports 1, 2 and 4 through 7 (as well as the expansion connector) available. If the 40/80 switch on the front of the computer is set to 80, then the port 3 area in the internal portion is always available, regardless of whether the rest of the internal portion is switched in.

The usual method for activating these ports from machine language is to load the port number into the Accumulator and call the routine at $FE95 to set an output port or $FE8B to set an input port. (See the Miscellaneous Routines listed earlier. These are equivalent to PR#n and IN#n.) $FE95 will store the actual address of the port driver code into the output port vector at locations $36,$37. $FE8B will store the address of the port driver code into the input port vector at locations $38,$39. Then to print a character through an output port, load the character to be printed into the Accumulator and call $FDED. This routine will actually jump "through" $36,$37 to the current output driver code. To read a character from a port, call $FD0C. This routine will jump through $38,$39 to the current input driver code.

Note: The port 1 printer routines and the port 3 80-column display routines will actually change the addresses stored in $36-$39 as part of the port initialization when they are first called. Some operating systems will also change the addresses stored here for internal bookkeeping reasons. Programs should normally call the $FDED and $FD0C routines to print and get characters through whatever port or operating system is "current", rather than access the ports directly.

For programs that do not use the System firmware at $F800 - $FFFF (for example, if the upper 16K RAM area is active), there are alternate entry points for ports 1,2,3, and 4. (But port 4 is a special case; see the section later on using the port 4 mouse firmware.) Ports 1, 2, and 3 contain special entry points to initialize the port, write characters to the port, read characters from the port, and determine the status of the port. These entry points are used by Pascal, newer versions of CP/M, and some application programs.

The initialization entry point needs to be called once, before reading from or writing to the port. After the read entry point is called, the character read will be returned in the Accumulator. Before the write entry point is called, the character to be printed should be loaded into the Accumulator. For the Status call, the Accumulator should contain a "0" to check write-ready status, or a "1" to check read-ready status. It will return either carry set meaning yes-ready, or carry clear meaning not-ready.

Port 1:

Init address	=	$C100 + contents of $C10D
Read address	=	$C100 + contents of $C10E
Write address	=	$C100 + contents of C10F
Status address	=	$C100 + contents of $C110

Port 2:

Init address	=	$C200 + contents of $C20D
Read address	=	$C200 + contents of $C20E
Write address	=	$C200 + contents of C20F
Status address	=	$C200 + contents of $C210

Port 3:

Init address	=	$C300 + contents of $C30D
Read address	=	$C300 + contents of $C30E
Write address	=	$C300 + contents of $C30F
Status address	=	$C300 + contents of $C310

10.2.1 Port 0: 40-column Display Routines

As the computer displays each character on the 40-column screen, the firmware maintains an invisible cursor position, marking where the next character will be printed. The horizontal position ($00 to $27 for 40-column) is kept in locations $24, and the vertical position ($00 to $17) is kept in location $25.

To set a new horizontal position directly, a program can store a new value into location $24. There are two ways to directly change the vertical position: Either load the new value into the Accumulator and call $FB5B, or store the value into location $25 and call $FC22.

The 40-column display routines normally print to the entire 40-column by 24-line display, though this can be changed. The video display routines (described earlier with the miscellaneous System routines) limit themselves to the current "text window". The text window is a rectangular area on the screen in which all printing takes place. Four zero-page locations determine the bounds of the text window:

$20 - horizontal position of left-edge of window
$21 - width of window (number of characters)

$22 - vertical position of top line in window.
$23 - vertical position of first line below bottom-edge of window

The routines at $FB2F and $FB39 set a full-screen 40 by 24 text window. A program can also change the text window values directly. The position of the left-edge added to the width should not exceed 40 for a 40-column display, or 80 for an 80-column display.

The left-edge of the window can be on either even or odd column, and the window width can be either an even or odd number of columns. If the window size or position is changed, the cursor should be placed inside of the window before any subsequent printing is done.

Besides displaying the normal printable characters, the port 0 output routines also recognize four printer control characters:

CTRL-G (ASCII $87) beeps the computer speaker.
CTRL-H (ASCII $88) moves the cursor left one position.
CTRL-J (ASCII $8A) moves the cursor down one line.
CTRL-M (RETURN, ASCII $8D) moves the cursor to the leftmost position and down one line.

When the port 0 input firmware is called to get a character from the keyboard, a checkerboard cursor blinks at the current cursor position. When a key is pressed, the original character is restored, and the ASCII value (high bit on) of the key pressed is returned in the Accumulator. An alternate entry point at $FD35 allows several special ESCape characters to be recognized from the keyboard. To use the ESCape codes, first press the ESCape key, then press the desired key:

ESC @ moves the cursor to the upper-left corner and clears the text window.

ESC E clears from the cursor position to the end of the line.
ESC F clears from the cursor position to the bottom of the text window.

The following keys can be pressed repeatedly after pressing ESCape:

ESC I or up arrow:	moves the cursor up one line.
ESC J or left arrow:	moves the cursor left one position.
ESC K or right arrow:	moves the cursor right one position.
ESC M or down arrow:	moves the cursor down one line.

10.2.2 Port 3: 80-column Display Routines

The 80-column routines are basically an enhancement of the 40-column routines. When port 3 is initialized, the video circuitry displays each character half as wide to allow up to 80 characters on each line. Locations $24 and $25 are still used to keep track of horizontal and vertical cursor position. Another location, $57B, also maintains the 80-column horizontal cursor position. Location $24, however, does maintain the horizontal position correctly for both 40 and 80 column displays on the computer.

The 80-column output firmware recognizes the same four control characters as the 40-column firmware, as well as several new ones. These control character are intended to be printed from a program. (Typing these characters directly at the keyboard may or may not cause them to be printed through the output firmware. For keyboard control, see the ESCape codes described below.)

CTRL-E: If using PASCAL, (alternate write-character entry point) "turns on" the visible cursor, displaying it as each character is printed. This is the usual setting for PASCAL.

CTRL-F: If using PASCAL, (alternate write-character entry point) "turns off" the visible cursor. Text display is faster with the visible cursor off.

CTRL-K: Clears the display from the cursor position to the bottom of the text window.

CTRL-L: Moves the cursor to the upper-left corner of the text window, and clears the entire text window.

CTRL-N: Displays subsequent text as "normal", white characters on a black background.

CTRL-O: Displays subsequent text as "inverse", black characters on a white background.

CTRL-Q: Switches to a 40-column display while keeping 80-column features.

CTRL-R: Switches back to an 80-column display.

CTRL-U: Switches to a 40-column display and turns off 80-column features.

CTRL-W: Scrolls the contents of the text window up one line, without moving the cursor.

CTRL-X: Do not display special graphics characters.

CTRL-Y: Moves cursor to upper-left corner of text window; does not clear text window.

CTRL-Z: Clears the entire line that the cursor is on; does not move the cursor.

CTRL-[: Displays special graphics characters instead of capital letters if "inverse" text is also set.

CTRL-\: Moves the cursor one position to the right.

CTRL-]: Clears from the cursor position to the end of the line.

CTRL- ^: Move the cursor up one line.

When the 80-column input routine is called to get a keypress, it displays an inverse-block cursor (the character at the cursor position is made inverse) while waiting for a keypress. If the alternate ESCape code entry point at $FD35 is called, additional ESCape codes become available from the keyboard. If the ESCape key pressed, the cursor changes to an inverse plus sign.

In addition to the 40-column ESCape codes, these codes are also recognized:

ESC 4: Switches to a 40-column display while keeping 80-column features.

ESC 8: Switches back to 80-column display.

ESC CTRL-Q: Switches to a 40-column display and turns off 80-column features.

ESC CTRL-D: Disables recognition of extra 80-column control character commands.

ESC CTRL-E: Enables recognition of extra 80-column control character commands.

10.2.3 Port 1: Printer Routines

The printer port is an output-only port. You can print characters through port 1 to a printer, but there is no provision for character input (since printers do not normally send characters back to a computer).

Besides simply sending the printed characters on to the printer, the port 1 firmware provides several other functions, depending on its current settings:

1. It will usually mask off the high bit of each character sent to a parallel printer (since many parallel printers use the high bit to represent some special code).

2. It can count the number of characters printed between each carriage return. If the number exceeds a given line width, it can send its own carriage return to the printer to start a new line.

3. It can follow any carriage return (whether printed by a program or added by the firmware) with a linefeed character, in case the printer needs one to begin printing on the next line down.

4. It can watch if the horizontal cursor position has been changed since the last character was printed (if an HTAB or BASIC command tab occurred), and send a series of spaces to the printer to accommodate the tab.

5. It can optionally echoes the text to the video display.

6. It can watch for special command character sequences for changing any of these features.

The Serial/Parallel switch on the front of the computer determines whether the serial or the parallel printer port is active. When the computer is first turned on, port 1 uses these settings:

80 characters per line
Insert linefeeds after carriage return
Do not add carriage return if line width is exceeded.
Do not echo characters back to the video display
Command character is a CTRL-I

Mask off high bit of each character (if parallel printer)
Format is 8 data bits, 2 stop bits (if serial printer)

Baud is 9600 (if serial printer)
Parity is set for No parity (if serial printer)

Several of these options can be changed by the Port Configuration Program, which is described in the computer User's Manual. The options set by the Port Configuration Program remain in effect as long as the computer is on. If you want to temporarily override those settings from within a program, you can issue individual command to the printer port. The override changes remain in effect until CTRL-RESET is pressed. Here are the command character sequences to print:

Parallel printer commands:

CTRL-I C: Add carriage return if line width is exceeded.

CTRL-I H: Send all 8 bits of each character to the printer. This is useful for some graphics printing.

CTRL-I I: Echo the text being printed back to the screen. This may cause problems for line widths other than 40 or 80 columns.

CTRL-I K: Don't automatically print a linefeed character after carriage return.

CTRL-I L: Automatically print a linefeed character after carriage return.

CTRL-I nnnN: Turn off screen echo and set the line width to nnn (where nnn is a number form 0 to 255). CTRL-I 0N sets "no line width" (do not insert carriage returns)

CTRL-I X: Send 7 (not 8) bits of each character to the printer. This is the usual setting.

CTRL-I Z: Do not check for any more commands.

CTRL-I "c": Change the printer command character from CTRL-I to the control character "c".

Serial printer commands:

CTRL-I nnB: Set the baud rate according to the number nn, where nn is:

1	50 baud
2	75 baud
3	110 baud
4	135 baud
5	150 baud
6	300 baud
7	600 baud
8	1200 baud
9	1800 baud
10	2400 baud
11	3600 baud
12	4800 baud
13	7200 baud
14	9600 baud
15	19200 baud

CTRL-I C: Add carriage return if line width is exceeded.

CTRL-I nD: Set the data format according to the number n, where n is:

0	8 data bits, 1 stop bit
1	7 data bits, 1 stop bit
2	6 data bits, 1 stop bit
3	5 data bits, 1 stop bit
4	8 data bits, 2 stop bits
5	7 data bits, 2 stop bits
6	6 data bits, 2 stop bits
7	5 data bits, 2 stop bits

CTRL-I I: Echo the text being printed back to the screen. This may cause problems for line widths other than 40 or 80 columns.

CTRL-I K: Don't automatically print a linefeed character after carriage return.

CTRL-I L: Automatically print a linefeed character after carriage return.

CTRL-I nnnN: Turn off screen echo and set the line width to nnn (where nnn is a number from 0 to 255). CTRL-I 0N sets "no line width" (do not insert carriage returns).

CTRL-I nP: Set the parity according to the number n, where n is:

0	no parity
1	odd parity
3	even parity
5	mark parity
7	space parity

CTRL-I Z: Do not check for any more commands.

CTRL-I "c": Change the printer command character from CTRL-I to the control-character "c".

10.2.4 Port 2: Serial Communications Routines

Port 2 is designed for two-way communication with modems, computer terminals, and other serial communications devices. A program can use port 2 to receive characters (after initializing with the routine at $FE8B or the BASIC command IN#2) or send characters (after initializing with $FE95 or PR#2), or both.

Since the serial input is not buffered, the firmware routines are best suited for communication at slower baud rates. For example, consider a program that gets characters from port 2 and prints them to the screen. Most of the screen display routines are quite fast, but scrolling operations take a little time. If too much time elapses between subsequent calls to read from port 2, one or more characters coming into the port may be missed or lost, because the program wasn't yet ready to receive them. For reliable high-speed communication, a modem program or terminal program that accesses the computer serial hardware directly is recommended.

Besides simply sending and receiving characters with port 2, the serial communications firmware provides several other functions, depending on its current settings:

1. It can follow any carriage return sent by the program with a linefeed character, in case the modem or terminal needs one at the end of every line.

2. It can optionally print the text to the video display.

3. It can watch for special command character sequences, for changing any of these features.

When the computer is first turned on, port 2 uses these settings:

Insert linefeeds after carriage return
Do not echo characters back to the video display
Command character is a CTRL-A
Format is 8 data bits, 1 stop bit
Baud is 300
Parity is set for No parity

As with the printer, several of the port 2 communications options can be changed by the Port Configuration Program. The options set by the Port Configuration Program remain in effect as long as the computer is on. If you want to temporarily override those settings from within a program, you can issue individual commands to port 2. The override changes remain in effect until CTRL-RESET is pressed. Here are the command character sequences to send:

CTRL-A nnB: Set the baud rate according to the number nn, where nn is:

1	50 baud
2	75 baud
3	110 baud
4	135 baud
5	150 baud
6	300 baud
8	1200 baud
9	1800 baud
10	2400 baud
11	3600 baud
12	4800 baud
13	7200 baud
14	9600 baud
15	19200 baud

CTRL-A nD: Set the data format according to the number n, where n is:

0	8 data bits, 1 stop bit
1	7 data bits, 1 stop bit
2	6 data bits, 1 stop bit
3	5 data bits, 1 stop bit
4	8 data bits, 2 stop bits
5	7 data bits, 2 stop bits
6	6 data bits, 2 stop bits
7	5 data bits, 2 stop bits

CTRL-A I: Echo the text being printed back to the screen. This may cause problems for line widths other than 40 or 80 columns.

CTRL-A K: Don't automatically print a linefeed character after carriage return.

CTRL-A L: Automatically print a linefeed character after carriage return.

CTRL-A nP:	Set the parity according to the number n, where n is:
0	no parity
1	odd parity
3	even parity
5	mark parity
7	space parity
CTRL-A Z:	Do not check for any more commands.
CTRL-A "c":	Change the command character from CTRL-A to the control-character "c".

CHAPTER 11
HARDWARE IMPLEMENTATION

11. HARDWARE IMPLEMENTATION

This chapter is written for those users who want to have a deeper understanding of the internal workings of the computer.

Besides providing valuable hardware information and trouble-shooting guidelines for service personnel, the materials described are also useful for hardware designers who need to know the characteristics and functions of the signals available at the various peripheral connectors on the computer.

11.1 System Overview

The computer uses advanced high-density CMOS gate array technology to reduce component count, interconnections, size of PCB and power consumption so that the reliability of the computer is improved.
Figure 11-1 and 11-2 shows the block diagrams of the "LASER 128" series computers.

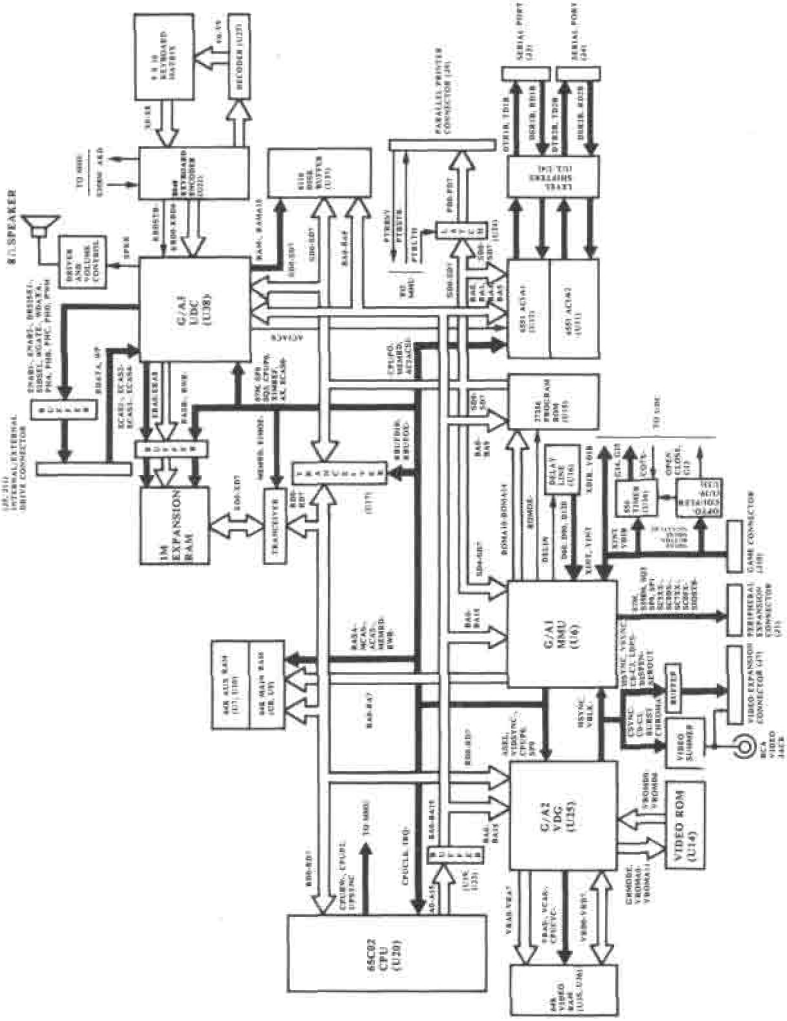


Fig. 11-1 Block diagram of "LASER 128EX"

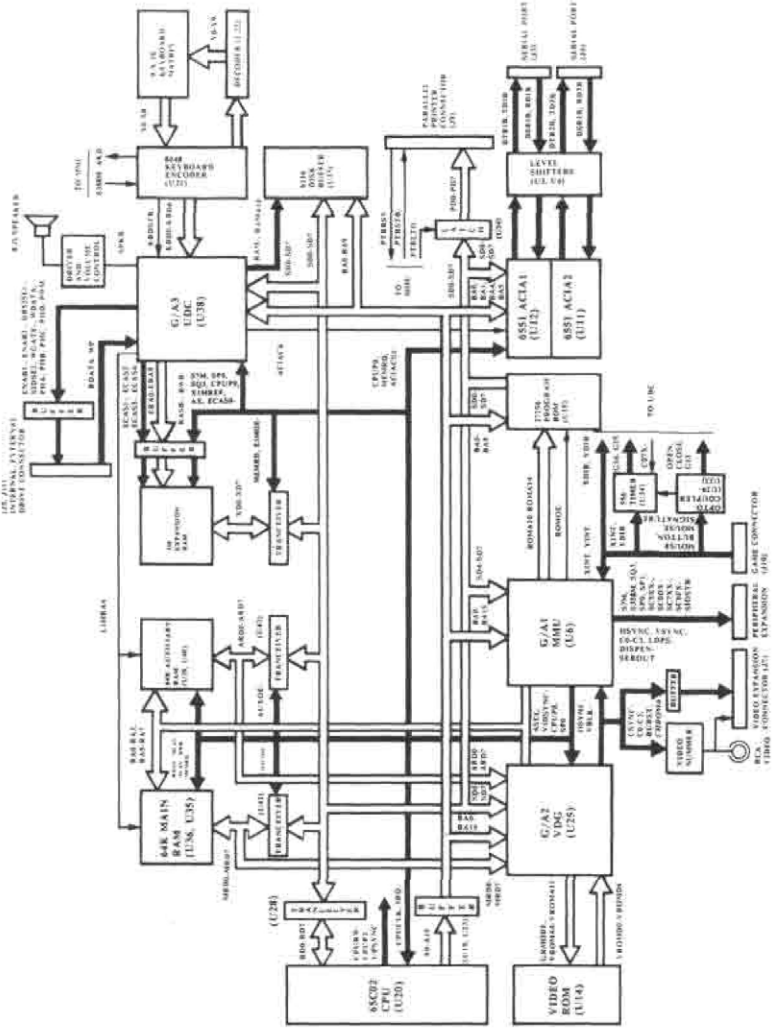


Fig. 11-2 Block diagram of "LASER 128"

The following is a brief description of the functions of the major components found on the main-board of the computer:

- Central Processing Unit (CPU): 65C02

  This is the microprocessor which is responsible for fetching and executing the machine language instructions stored in the program memory (either ROM or RAM).

  The 65C02 used in "LASER 128EX" is a 4 MHz version while that used in "LASER 128" is a 1 MHz version.

- Memory Management Unit (MMU): 61H20B48F

  This is one of the three custom-designed gate arrays used in the computer. Its primary function is to control the addressing and timing of the memory devices (RAM and ROM). Besides, it also contains the logic for generating the various system clock signals as well as controlling the operation of some of the I/O devices such as mouse, parallel printer and serial ports.

- Video Display Generator (VDG): 61H20B44F

  This is another custom-designed gate array which is responsible for generating and maintaining the video display.

  It fetches and decodes the data stored in the video RAM to generate the digital video outputs, some of which is accessible at the video expansion connector on the back panel of the computer. They are also processed by analog circuitries to produce the composite video output.

- Universal Disk Controller (UDC): 61H20B56F

  This gate array is primarily used for controlling the operation of the built-in or external disk drives connected to the computer.

  Besides, it also controls the addressing of the expansion RAM and the operation of the other I/O devices such as keyboard and speaker.

- Program ROM: 27C256

  It is a 32K x 8-bit ROM chip which contains the built-in BASIC interpreter, monitor and firmware drivers for the various I/O devices.

- Video ROM: VT27-0706-0

  This is a 8K x 8-bit mask ROM which contains the text display character fonts and part of the graphics display generation logic. It works together with the VDG gate array to generate the video display.

- System RAM: 41464 (X 4)

  They are 64K x 4-bit dynamic RAM chips which are organized into two 64 K-byte banks, namely main and auxiliary bank. The RAM chips used in the "LASER 128 EX" have a maximum row address access time of 120 ns while those used in the "LASER 128" have a row address access time smaller than 150 ns.

- Video RAM (for "LASER 128 EX" only): 41464 (X 2)

  They are used for storing the memory image of the video display in a specially encoded format. The maximum row address access time for the RAM chips is 150 ns.

  The VDG gate array fetches and decodes the contents of the video RAM in regular time intervals to generate the appropriate video signals to refresh the video display. In "LASER 128", the system RAM is used for generating the video display so that no separate video RAM is required.

- Expansion RAM (Optional): 41256 (x 32)

  These 256K x 1-bit dynamic RAM chips have a maximum row address access time of 120 ns. They are inserted in the IC sockets on the optional 1M RAM expansion card.

  In "LASER 128 EX", the expansion RAM card is built-in. A maximum of 1 M-byte expansion RAM can be installed if all of the 32 IC sockets on the RAM board are fully stuffed.

- Disk data buffer: 6116

  This 2K x 8-bit static RAM serves as a temporary CPU working area for storing the data read from or written to a 3.5" disk drive as well as other system parameters used by the internal port 7 firmware.

- Keyboard encoder: 8048

  It is a single-chip microcontroller with internal ROM, RAM, I/O ports etc. and is specially programmed to handle the task of keyboard interfacing.

  It scans the keyboard for any keypress, generates the appropriate character code and stores it in a register in the UDC gate array for subsequent read-out by the CPU.

- Asynchronous Communication Interface Adaptor: 6551 (x2)

  It is the IC which handles the data transfer between the computer and the devices connected to the two serial interface connectors on the back panel of the computer. There are two ACIA's on the main-board, one for each of the serial ports.

It performs such functions as parallel-to-serial and serial-to-parallel data conversions, parity generation for transmit data and error detection for receive data, data buffering and communication handshaking etc..

11.2 Clock generator

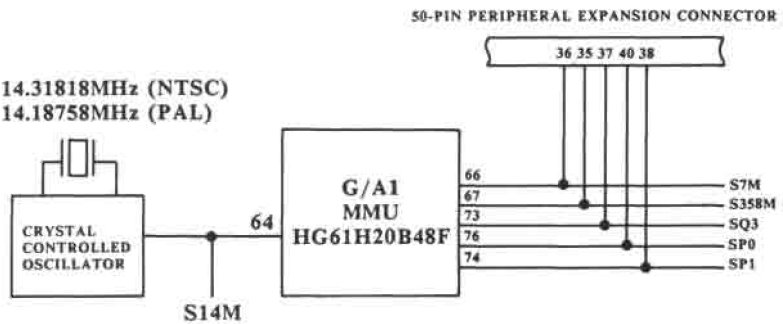


Fig. 11-3 Clock generator

The "heart" of the computer is a crystal controlled oscillator operating at a frequency of 14.31818 MHz (for NTSC models) or 14.18758 MHz (for PAL models). The output of the oscillator is buffered to become signal S14M which is the master clock to which all system activities are synchronized.

S14M is fed into the MMU gate array which is then divided down to generate the general purpose clock signals S7M (7 MHz), S358M (3.58 MHz), SQ3 (2 MHz), SP0 and SP1 (1 MHz). The timing diagram is shown in Fig. 11-4.

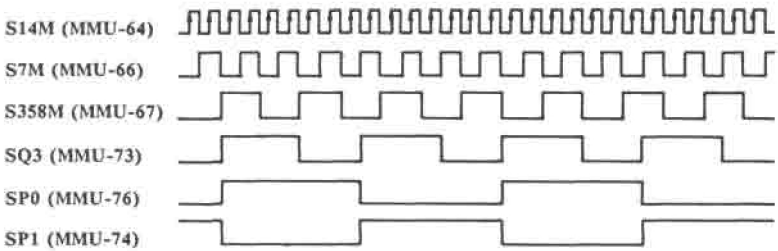


Fig. 11-4 Timing diagram of general purpose clocks

11.3 CPU and system buses

The 65C02 CPU has a 16-bit address bus and a 8-bit data bus. The CPU address bus (A0 - A15) is buffered by two 74HCT244 (octal buffers) to become the system address bus (BA0 - BA15) which is routed to the MMU, VDG, UDC, 27256, 6116, 6551 and the expansion connector.

The CPU data bus (RD0 - RD7) is buffered by a 74HCT245 (octal bus transceivers) to become the system data bus (SD0 - SD7). The organization of the data bus in "LASER 128 EX" is different from that of "LASER 128".

In "LASER 128", the CPU data bus is completely isolated from the other components by the data bus buffer. The system data bus is routed to the MMU, VDG, UDC, 27256, 6116, 6551, parallel printer data latch (74HCT374), expansion RAM connector and peripheral expansion connector.

The system data bus is also isolated from the main bank RAM data bus (MD0 - MD7), auxiliary bank RAM data bus (AD0 - AD7) and the expansion RAM data bus (XD0 - XD7) by octal bus transceivers (74HCT245).

In "LASER 128 EX", the CPU data bus is directly connected to the VDG, the system RAM data bus and the expansion RAM data bus buffer. The system data bus is connected to the MMU, UDC, 27256, 6116, 6551, parallel printer data latch and the peripheral expansion connector.

The high-order bit of the system address bus (BA15) and the CPU read/write signal (CPURW-) is isolated from the peripheral expansion connector by the MMU gate array.

When signal DMA- on the peripheral expansion connector (pin 22) is pulled low by an interface card plugged into the optional expansion box, the system address bus, system data bus and read/write line will enter high impedance state so that the external interface card can control the system buses to access the memory and I/O devices directly without CPU intervention.

The clock signal for the CPU (CPUCLK) is generated by the MMU gate array (pin 34).

In "LASER 128", this signal has the same timing as SP0 so that the CPU will always run at 1 MHz.

In "LASER 128 EX", CPUCLK may have different frequencies and waveforms at different instants. Depending on the states of the internal registers and the address being accessed, the CPU may run at 1 MHz, 2.3 MHz or 3.6 MHz. The timing waveforms of the CPU clock at various frequencies are shown in Fig. 11-5.

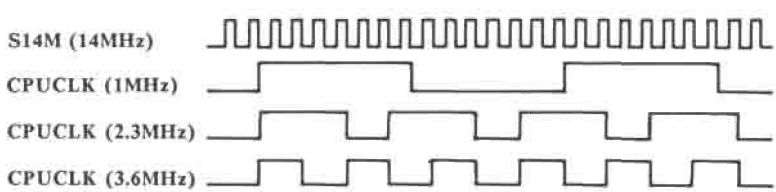


Fig. 11-5 Timing diagram of CPU clock

When an external interface card plugged into the expansion box pulls low signal DMA- to perform direct memory access, CPUCLK will also be forced low at the same time to stop the CPU. Notice that DMA- should only switch state when CPUCLK is low. Otherwise, the current bus cycle may not terminate properly.

Moreover, since only SP0 (but not CPUCLK) is available at the peripheral expansion connector, direct memory access can only be performed properly when the CPU is running at 1 MHz, during which CPUCLK is equivalent to SP0.

11.4 Program ROM control

The program ROM is divided into 32 pages, each of which is 1 K-byte in size.

The 10 low-order address inputs of the ROM are wired directly to the system address lines (BA0 to BA9). The MMU gate array re-maps the contents of the system address bus according to the settings of internal registers to generate the five high-order ROM address inputs (ROMA10 to ROMA14).

Fig. 11-6 shows the physical address map of the ROM.

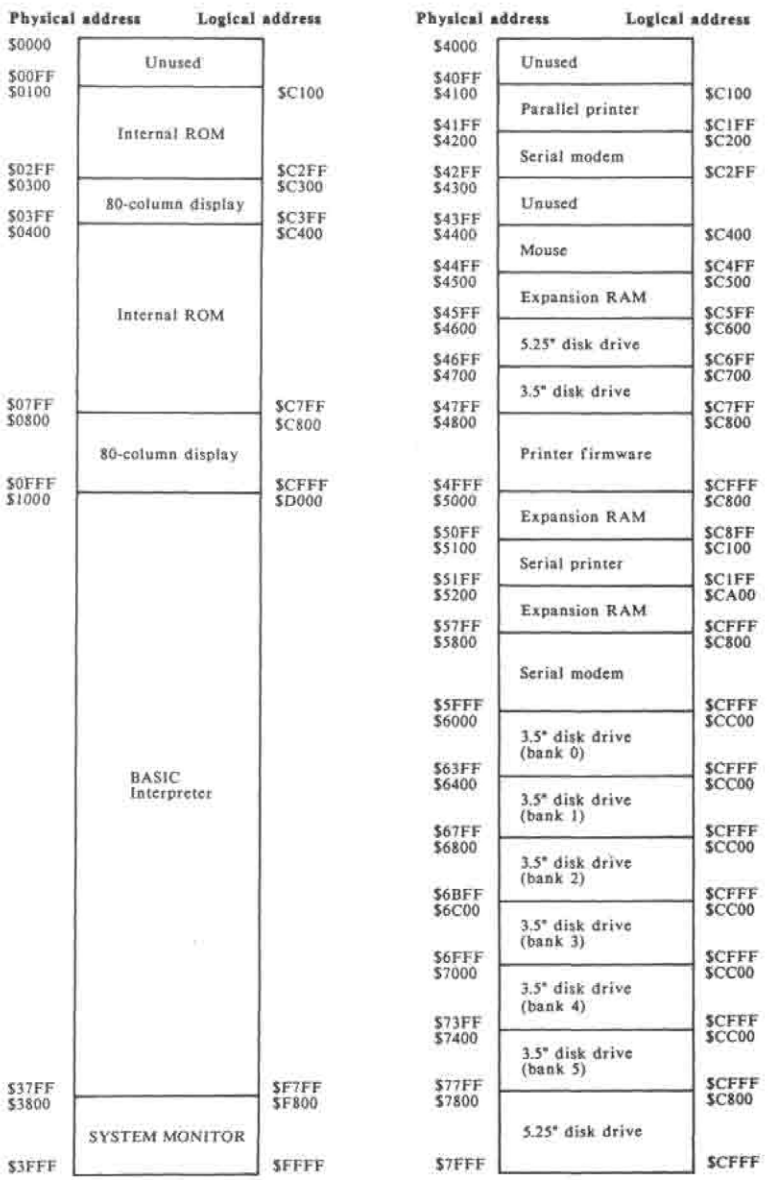


Fig. 11-6 Program ROM address map

The MMU gate array also decodes the contents of the system address bus to generate the ROM output buffer control signal (ROMOE-). Signal CPUP0 (MMU-40), which is the same as CPUCLK except that it is not gated with DMA-, is used for qualifying the validity of the contents of the address lines.

If the address being accessed falls into the range which is assigned as ROM space, the MMU gate array will pull low ROMOE- as CPUP0 goes high, driving the contents of the addressed ROM location into the system data bus. The bus master (65C02 or external interface card) should strobe in the contents of the system data bus at the falling-edge of CPUP0.

11.5 System RAM control

The system RAM is divided into two 64K-byte banks, namely main and auxiliary.

The MMU gate array multiplexes a re-mapped version of the system address bus and the outputs of the internal dynamic RAM refresh address generator into eight RAM address lines (RA0 to RA7) which are connected to both the main and auxiliary RAM banks.

In "LASER 128 EX", the MMU gate array uses an external delay line to generate the RAM timing signals in the two high-speed modes (2.3 MHz and 3.6 MHz).

The delay line input (DELIN) is derived from the clock output of 65C02 (CPUP2). The delay line outputs D60, D90 and D120 are routed back into the MMU gate array as timing references for the signals RASA-, AX, MCAS-, ACAS- and RWR-.

RASA- is connected to the row address strobe (RAS-) input of the dynamic RAM chips. At the falling-edge of this signal, the contents in the multiplexed RAM address lines RA0 to RA7 will be strobed into the dynamic RAM chips as row address.

AX is looped back into the MMU gate array through pin 32 to control the multiplexing of the dynamic RAM address. When this signal is high, row address will appear on the RAM address bus RA0 to RA7. When this signal is low, column address will be selected.

MCAS- and ACAS- is connected to the column address strobe (CAS-) input of the main and auxiliary RAM bank respectively. At the falling-edge of this signal, the column address on the RAM address bus will be strobed into the RAM chips.

The RWR- signal is connected to the write-enable input of the dynamic RAM chips. If RWR- is high, a read operation will be performed in the current RAM cycle. If it is low, the current RAM cycle will perform a write operation.

Signal MEMRD- is connected to the output buffer control input of the RAM chips. In a RAM read cycle, MEMRD- will be asserted (goes low) when CPUP0 is high in order to drive the contents of the addressed RAM location into the RAM data bus.

In the "LASER 128" or "LASER 128 EX" running at 1 MHz, all the dynamic RAM timing signals are generated internally, without using the delay line outputs.

Fig. 11-7 shows the timing diagrams of the dynamic RAM control signals at various CPU clock speeds.

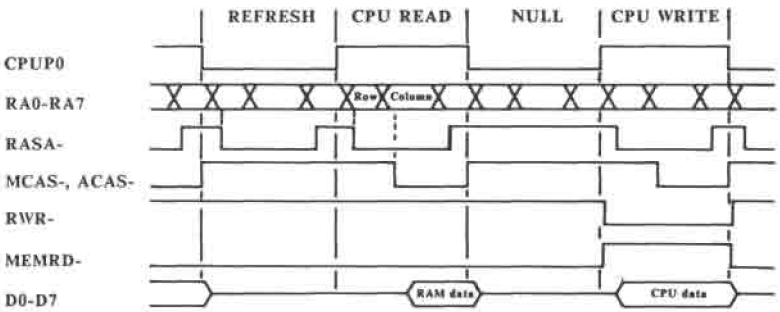
In "LASER 128 EX", the system RAM address bus (RA0-RA7) is isolated from the video RAM address bus (VRA0-VRA7). The main and auxiliary RAM data bus are wired together and connected directly to the CPU data bus RD0 - RD7.

In "LASER 128", the system RAM address bus and video RAM address bus are wired together . RA4 is remapped by the UDC gate array into L1MRA4 before connecting to the RAM chips. The rest of the RAM address lines are connected directly to the RAM chips.

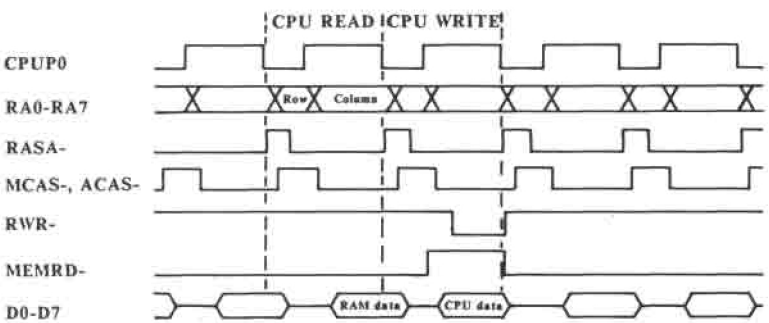
Moreover, the main and auxiliary RAM data bus are isolated from each other and the system data bus SD0-SD7 by octal bus transceivers. Signals MAINOE- and AUXOE- from the MMU gate array are connected to the output control of the main and auxiliary RAM data bus buffer respectively.

When the main RAM bank is being accessed, signal MAINOE- will go low when CPUP0 is high to enable the output buffers of the main RAM data bus transceiver. Similarly, AUXOE- will be asserted when the auxiliary RAM bank is being accessed.

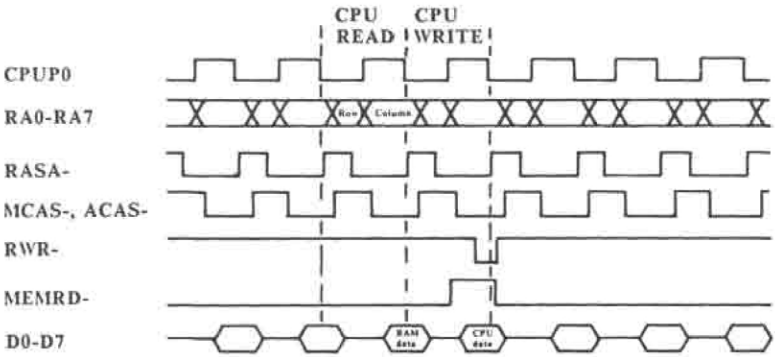
The two RAM data buses (MRD0-MRD7, ARD0-ARD7) are also connected to the VDG gate array. The video data stored in the system RAM is transferred to the VDG gate array 16 bits at a time. This will be revisited later in the section devoted to video display generation circuit.



(a) 1 MHz CPU clock



(b) 2.3 MHz CPU clock



(c) 3.6 MHz CPU clock

Fig. 11-7 System RAM timing diagrams (LASER 128 EX)

11.6 Video display generation

The video display generation hardware is made up of the VDG gate array, video RAM, video ROM as well as some "glue" logic and analog circuitries.

In "LASER 128 EX", there is a dedicated 64 K-byte video RAM for storing the video data. The video RAM is time-multiplexed between the CPU and the video generation circuitries.

The VDG gate array decodes the contents in the system address bus BA0 - BA15 and the system read/write line R/W- to determine if the CPU or the current bus master wants to perform a write operation on the video RAM. If so, the contents in the system address bus, CPU data bus and the signal ASEL (high when accessing auxiliary RAM bank) from the MMU gate array will be latched inside the VDG gate array at the falling-edge of signal CPUP0 for subsequent use.

The VDG gate array multiplexes the outputs of the CPU address latch with the those of the internal video address generator and refresh address generator into the video RAM address bus VRA0 to VRA7 using the signals CPUCYC- and HSYNC.

When signal CPUCYC- is low, the outputs of the CPU address latch is selected. When CPUCYC- is high and HSYNC is low, the outputs of the video address generator is selected. When CPUCYC- and HSYNC are both high, the outputs of the refresh address generator will be selected.

Signal CPUCYC- is also connected to the write enable input of the video RAM. If it is low, write operation will be performed in the current RAM cycle. Otherwise, either a video read cycle or a refresh cycle will be performed.

The row address strobe (VRAS-) and column address strobe (VCAS-) for the video RAM is also generated by the VDG gate array. VRAS- will be asserted (low) at either a CPU write, video read or refresh cycle whereas VCAS- will only be asserted at a CPU write cycle or video read cycle.

The video RAM data bus VRD0-VRD7 is connected directly to the VDG gate array. In a CPU write cycle, the VDG gate array will drive the outputs of the CPU data latch into the video RAM data bus when signal CPUCYC- is low. In a video read cycle, the contents of the addressed location will be driven into the video RAM data bus after VCAS- goes low. The video data will then be latched in the VDG gate array.

Fig. 11-8 shows the timing diagram of video RAM read/write operations in "LASER 128 EX".

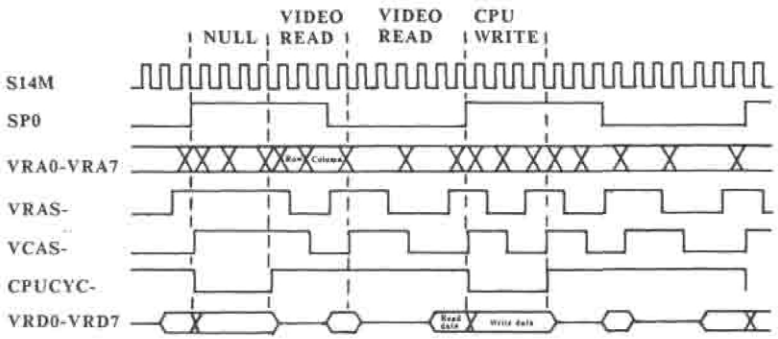


Fig. 11-8 Video RAM timing diagram for "LASER 128 EX"

In "LASER 128", part of the system RAM is reserved for the video display so that no dedicated video RAM is required as in "LASER 128 EX".

The video RAM address bus VRA0-VRA7 and the system RAM address bus RA0-RA7 are wired together and connected to the system RAM. The RAM address bus is time-multiplexed between the MMU gate array and the VDG gate array using the signals SP0 and HSYNC.

When SP0 is high, the MMU gate array multiplexes the contents of the system address bus into the system RAM address bus. When SP0 and HSYNC are both low, the VDG gate array multiplexes the contents of the video address generator into the system RAM address bus. Finally, if SP0 is low and HSYNC is high, the MMU gate array will drive the outputs of the internal refresh address generator into the RAM address bus.

The system RAM control timing signals (RASA-, MCAS-, ACAS-, RWR-, MEMRD-) are generated by the MMU gate array as described before.

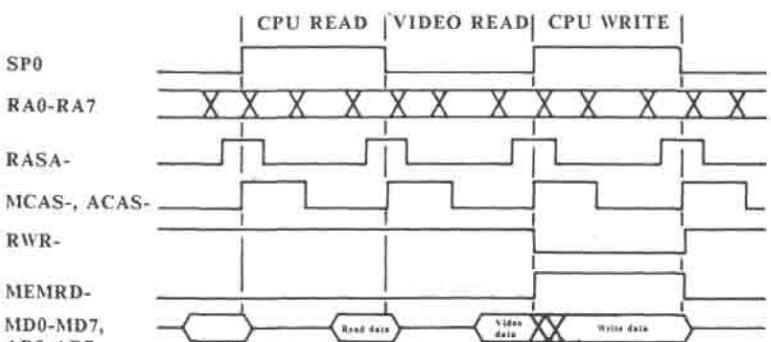


Fig. 11-9 System RAM timing diagram for "LASER 128"

The main RAM data bus MRD0-MRD7 and auxiliary RAM data bus ARD0-ARD7 are connected directly to the VDG gate array. At the rising-edge of SP0, the data on the two RAM data buses are latched into the VDG gate array simultaneously.

In conclusion, two bytes of video data are transferred from the video RAM into the VDG gate array during each SP0 cycle. In "LASER 128 EX", this is performed in two consecutive video RAM cycle whereas in "LASER 128", only a single RAM cycle is required to accomplish the same video data transfer rate.

The VDG gate array uses the latched video RAM data together with the states of the internal registers and position counters to generate the 13 address inputs to the video ROM, namely VROMA0 to VROMA11 and GRMODE.

GRMODE is a signal which indicates to the video ROM which display mode the computer is currently in. When it is low, the display is in text mode. Otherwise, graphics will be displayed. The signals sent to the video ROM through VROMA0 to VROMA11 are different for the various display modes.

In text mode, i.e. GRMODE is low, the video ROM is used as a character generator, storing the character fonts in a bit-mapped pattern. There are two sets of character fonts available, selected by the USACG-input of the VDG gate array (pin 16). This pin is tied either high or low for a particular version of the computer. For instance, this pin is grounded in NTSC versions.

Each character font occupies eight contiguous bytes in the video ROM. Each of the bytes is mapped to one of the rows of the 7Hx8V dot matrix for that character, with the first byte mapped to row 0, i.e. the uppermost row.

Bit 0 to bit 6 of each byte are mapped to the seven dots in the corresponding character row, with bit 0 mapped to the leftmost dot. Bit 7 is not used. A zero bit corresponds to a black dot while a one bit corresponds to a white dot.

To display a particular row of a character font, the VDG places the 8-bit character code on VROMA3 to VROMA10 and the 3-bit row number on VROMA0 to VROMA2 and then read back the bit patterns of that particular character row from the video ROM through VROMD0 to VROMD6. The bit patterns will then be shifted out through COLOR0 to COLOR3.

In graphics mode, i.e. GRMODE is high, the video ROM is used as a block of slow combinational logic network having 12 inputs (VROMA0 to VROMA11) and 7 outputs (VROMD0 to VROMD7).

Fig. 11-10 shows a simplified address map of the video ROM.

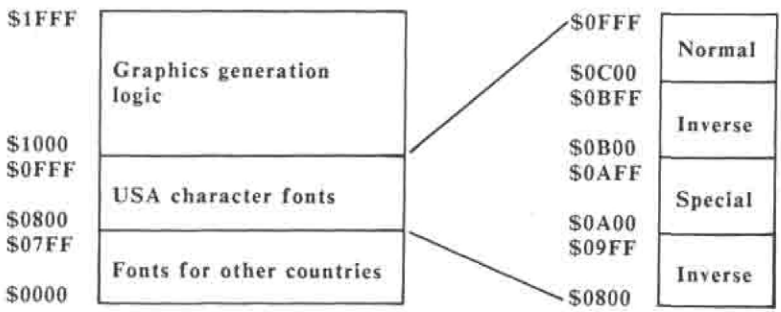


Fig. 11-10 Address map of video ROM

The VDG gate array encodes the video ROM outputs into 4-bit color codes at COLOR0 to COLOR3. Depending the current graphics mode, the color encoding scheme may be different so that the same video ROM outputs may be encoded into different colors in different graphics mode.

Besides color information, the VDG is also responsible for generating the video display synchronisation signals HSYNC (pin 51), VSYNC (pin 52) and CSYNC- (pin 48).

HSYNC is the horizontal synchronisation signal for the video display which is connected to pin 9 of the video expansion connector. The rising-edge of this signal triggers the horizontal retrace of the electron beam in the video monitor. The timing of HSYNC is shown in Fig. 11-11.

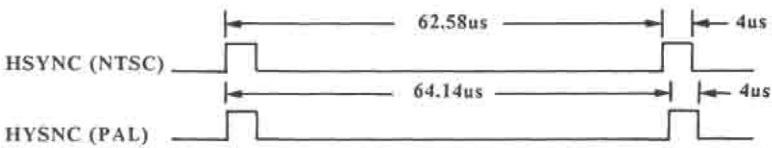


Fig. 11-11 Timing of HSYNC

VSYNC is the vertical synchronisation signal which is connected to pin 14 of the video expansion connector. A high level at this signal triggers the vertical retrace of the electron beam in the video monitor. Its timing is shown in Fig. 11-12.

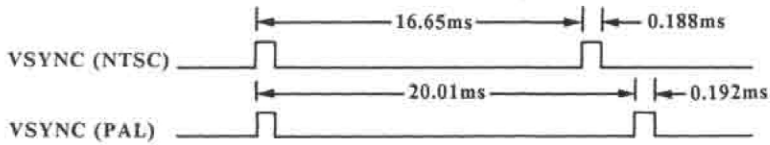


Fig. 11-12 Timing of VSYNC

CSYNC- is the composite synchronisation signal which contains both horizontal and vertical synchronisation pulses. It is low if either HSYNC or VSYNC is high and is high otherwise.

The color information carried in the composite video signal is originated from the VDG outputs BURST (pin 49) and CHROMA (pin 50).

After each horizontal synchronisation pulse, a short burst of several cycles in duration will be generated at BURST. This color burst has a frequency equal to that of the color subcarrier for the particular television system adopted. For NTSC system, this will be 3.58 MHz whereas for PAL systems, this will be 4.43 MHz. The timing is shown in Fig. 11-13.

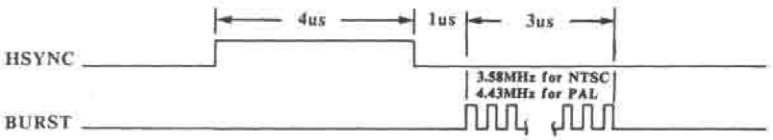


Fig. 11-13 Color burst timing

The function of the color burst is to phase-lock the internal oscillator in the composite color monitor or TV, thus generating a precise phase reference.

CHROMA has the same frequency as the color burst. However, it may have different phases depending on the color code outputs at COLOR0 to COLOR3. It is this phase information embedded in a composite video signal which enables different colors to be generated by the color monitor or TV.

A circle diagram is usually used for showing the relationship between phase and color. In this diagram, the circumference of the color circle is labelled with phase angles and the corresponding colors. The phase angles are measured in clockwise direction. Larger phase angle corresponds to larger phase lag or delay.

For NTSC system, the phase angle of the color burst is taken as 0 degree which corresponds to yellow. For example, red corresponds to a phase angle of 90 degrees and hence the chroma for red lags the color burst by 90 degrees. Fig. 11-14 shows the NTSC color circle.

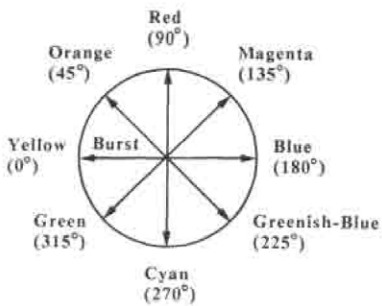
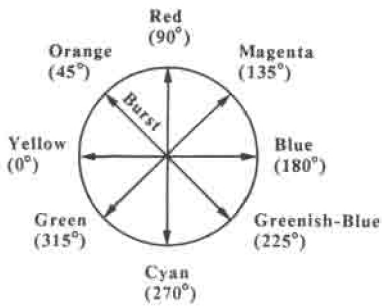


Fig. 11-14 NTSC color circle

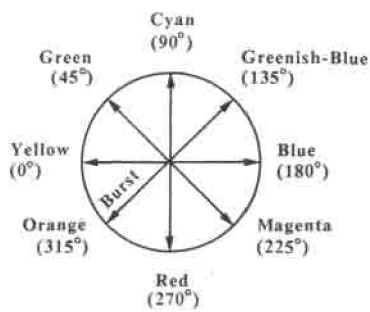
For PAL system, the color circle is reversed every other line. For example, red is +90 degrees for a line and +270 degrees (or -90 degrees) for another line.

The line which has the same color circle as NTSC system is referred to as NTSC line. The color burst of NTSC line is taken to be 45 degrees, which corresponds to orange. For example, red corresponds to a phase angle of 90 degrees so that the chroma for red in a NTSC line lags that of the color burst by 45 degrees.

The line which has a reversed color circle is referred to as PAL line. The color burst of PAL line is taken to be 315 degrees (or -45 degrees). Red corresponds to a phase angle of 270 degrees (or -90 degrees) so that the chroma for red in a PAL line leads that of the color burst by 45 degrees.
Fig. 11-15 shows the color circles for PAL system.



NTSC line



PAL line

Fig. 11-15 PAL color circles

The frequency reference for BURST and CHROMA is generated by the VDG gate array by dividing the input XTAL (pin 13) by four.

For NTSC system, XTAL is connected to S14M which has a frequency of 14.31818 MHz so that the color subcarrier frequency is 3.58 MHz.

For PAL system, the signal S358M (pin 67 of MMU) generated by dividing S14M by four, is buffered to become the input to a tune amplifier having a centre frequency of 17.73447 MHz. The fifth harmonic of the buffered input is extracted to become signal S17M which is connected to the input XTAL of the VDG. As a result, the signal S17M and hence CHROMA and BURST is phase-locked to S14M.

When the "COLOR/MONO" switch is thrown to "MONO" position, the BURST and CHROMA output will be disabled (low) so that color is not available. The color burst is also disabled in text mode so as to remove any unwanted color fringes from the characters, thus producing a better display.

The signals COLOR0 to COLOR3, BURST, CHROMA and CSYNC- are combined by a transistor mixer with resistor-summing inputs to become the composite video signal which is connected to both the RCA jack and pin 12 of the video expansion connector on the back panel of the computer.

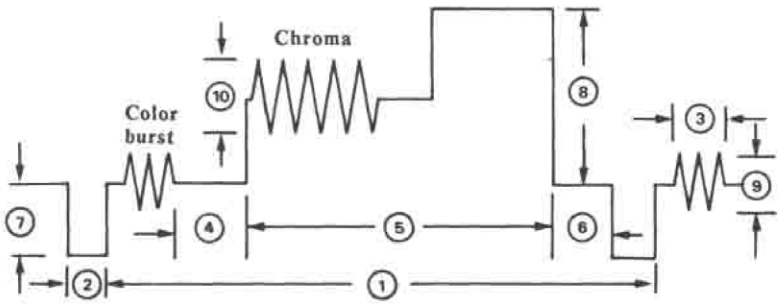
COLOR0 to COLOR3 are d.c. coupled to the transistor mixer. The resistor summing network at the base of the transistor acts as a simple digital-to-analog converter, converting the 4-bit color code into an appropriate voltage level. This voltage component in the composite video signal is called luminence. The brightness of a particular spot on the screen is determined by the instantaneous value of this parameter. The summing resistors for COLOR0 to COLOR3 have the same resistance so that only four levels of brightness are available.

The CSYNC- signal is also d.c. coupled to the video summer. During the active display interval, CSYNC- is always high. As a result, the luminence signal is superimposed on the blanking level.

BURST and CHROMA are a.c. coupled to the transistor mixer. Since the color burst is generated during the horizontal blanking interval (after the horizontal sync pulse), it will be centred at the blanking level.

CHROMA will always be low during the horizontal and vertical blanking interval. When black, gray or white is being displayed, CHROMA will be a constant d.c. level which will be blocked by the coupling capacitor. For the other colors, the chrominence signal will be centred at the d.c. level of the luminence signal.

The composite video output is designed for driving a 75-Ohm load. The peak-to-peak voltage level of the composite video output is adjusted to about 1 V. Fig. 11-16 shows the waveform and the relative levels of the various components of the composite video signal.



Parameter	Description	NTSC	PAL
1	Horizontal line period	62.6us	64.1us
2	Horizontal sync pulse width	3.91us	3.95us
3	Color burst length	2.93us	2.96us
4	Color back porch	5.87us	5.92us
5	Horizontal active display	39.1us	39.5us
6	Front porch	9.81us	10.78us
7	Sync level	0.4V	0.4V
8	White level	0.8V	0.8V
9	Peak-to-peak burst level	0.26V	0.26V
10	Peak-to-peak chroma level	0.6V	0.6V

Fig. 11-16 Composite video signal

The VDG gate array also generate the LCD panel control signals DISPEN- (pin 53), LDPS- (pin 54) and SEROUT (pin 55).

DISPEN- is effectively a composite blanking signal. During the horizontal and vertical blanking interval, DISPEN- is high. During the active display interval, it is low

LDPS- is a periodic timing synchronization signal. When 40-column text, low-resolution graphics or high-resolution graphics is being displayed, it will be a 1 MHz clock signal having a duty cycle of 13H:1L.

When 80-column text, double-low-resolution graphics or double-high-resolution graphics is being displayed, LDPS- will be a 2 MHz signal having a duty cycle of 6H:1L. It is delayed by a S14M cycle before connecting to pin 6 of the video expansion connector.

SEROUT is the serial video data for the LCD panel. It is just COLOR0 delayed by one S14M cycle. SEROUT is inverted and delayed by another S14M cycle using an external D-type flip-flop before connecting to pin 11 of the video expansion connector.

Fig. 11-17 shows a snap-shot of the timing waveforms of the LCD display control signals at the start of a white line on the screen.

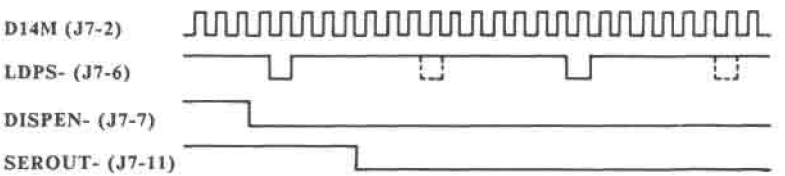


Fig. 11-17 Timing of LCD panel control signals

11.7 Keyboard control

The keyboard controller is made up of the 8048 keyboard encoder, 74LS145 BCD (Binary-coded-decimal) decoder, UDC gate array and MMU gate array. Figure 11-18 shows the simplified block diagram of the keyboard controller.

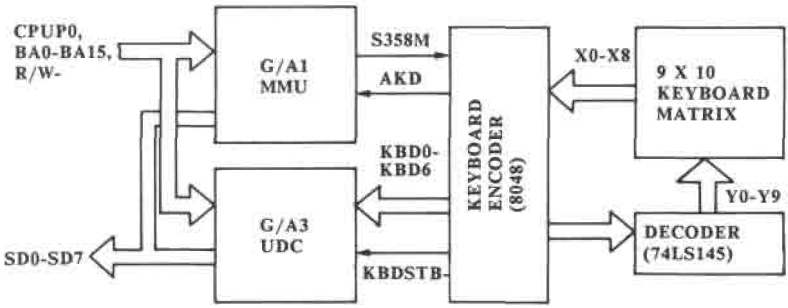


Fig. 11-18 Block diagram of keyboard controller

The 8048 is a single-chip microcomputer with internal ROM, RAM and I/O ports. It is clocked by the signal S358M from the MMU gate array and hence runs at 3.58 MHz.

Four of the output port bits (P20-P23) of the keyboard encoder are decoded by the BCD decoder 74LS145 into 10 signal lines which are connected to the Y-lines of the 9x10 keyboard matrix (Y0 to Y9). The X-lines of the keyboard matrix (X0 to X8) are connected to the nine input port bits P10 to P17 and P24 of the keyboard encoder.

The keyboard encoder cycles the output port bits P20-P23 from 0 to 9 continuously so that the decoder outputs will be driven low one-by-one sequentially. If any key on the keyboard is pressed, one of the X-lines will be pulled low when the appropriate Y-line is driven low. The keyboard encoder will sense the occurrence of a keypress by reading the input port bits and assert the AKD (Any-Key-Down) signal which is connected to pin 22 of the MMU gate array. The state of this signal can then be read back by the CPU at bit 7 of the I/O location $C010.

The location of the key being pressed can be identified by the states of the X-lines and Y-lines. The keyboard encoder converts this information into a key code using internal mapping ROM and the status of the "STD/ALT" switch. It then places the key code on the data lines KBD0 to KBD6 and signals to the UDC gate array that valid key code is available on the data lines by generating a low pulse at the signal KBDSTB-.

At the rising-edge of KBDSTB-, the key code is latched into the UDC gate array. A flip-flop in the UDC gate array is also set to indicate that a valid key code has been received. When the CPU performs a read operation on the I/O location $C000, the 7-bit key code and the valid-key flag will be driven into the system data bus and strobed in by the CPU.

11.8 Speaker control

Whenever the location $C030 is accessed, the UDC gate array output SPKR (pin 21) will be toggled. It is amplified by transistor circuits to increase the driving power before connecting to the built-in 8-Ohm speaker. By varying the time interval between successive accesses to the location $C030, waveforms of different frequencies can be generated at SPKR and hence tones of different pitches can be produced.

11.9 Disk drive control

All the disk drive control signals are originated from the UDC gate array.

The UDC output DR35 (pin 55) is high if 3.5" disk drive is being selected and is low otherwise.

The signal ENAB1- (pin 12) will be low if the locations $C0E9 and $C0EA is accessed. It will be high if either $C0E8 or $C0EB is accessed. It is buffered and connected to pin 9 of the external drive connector.

INT525EN- (pin 56) is buffered and connected to pin 14 of the internal drive connector on the main PCB. It will go low if both DR35 and ENAB1- is low and will be high otherwise. When low, the disk drive attached to the connector will be enabled.

ENAB2- (pin 13) will be low if the locations $C0E9 and $C0EB is accessed. It will be high if either $C0E8 or $C0EA is accessed. It is separately buffered and connected to pin 4 and pin 17 of the external drive connector. When low, the disk drive connected to the external drive connector will be enabled.

The signals PHA (pin 3), PHB (pin 5), PHC (pin 6) and PHD (pin 7) are buffered and connected to both the internal and external connectors. They can be set high by accessing the I/O locations $C0E1, $C0E3, $C0E5 and $C0E7 and set low by accessing the locations $C0E0, $C0E2, $C0E4 and $C0E6 respectively.

These signals control the voltages applied to the four phases of the stepper motor in a 5.25" disk drive. When high, voltage is applied to the corresponding stepper phase. When low, no voltage is applied. To step the read/write head, the four phases are turned on one-by-one sequentially. In addition, they are also used for sending high-level commands to intelligent disk drives and 3.5" drives.

SIDSEL (pin 11) is for selecting one of the two disk surfaces for read/write in a double-side drive. It is buffered and connected to pin 9 of the internal drive connector and pin 16 of the external drive connector. To set it high, write a one to bit 7 of location $C0ED while both ENAB1- and ENAB2- are high. Writing a zero to the same bit turns SIDSEL low.

The signal WGATE- (pin 10) is buffered to the drive connectors. When the CPU accesses the I/O location $C0EF, this signal will go low to enable the write circuitries in the disk drive. On the other hand, if the CPU accesses the location $C0EE, WGATE- will go high so that the disk drive will be in read mode.

WDATA (pin 9) is the serialized write data for the disk drive. It is buffered to pin 18 of the internal drive connector and pin 19 of the external drive connector.

PWM is buffered by an open-collector inverter and connected to pin 10 of the external drive connector. It is a pulse-width modulated signal having a constant frequency but varying duty cycle for controlling the speed of the spindle motor in a 3.5" drive.

WP (pin 17) is the write-protect signal from a 5.25" disk drive. It is valid only when the disk drive is enabled. When high, it indicates that a write-protected diskette is inserted in the disk drive. This signal shares the same pin with PWM on the two drive connectors.

RDATA (pin 8) is the read data from the disk drive. Whenever the drive is enabled, positive read pulses with be generated at this signal line. For Macintosh type 3.5" disk drive, it is also the signal path through which the drive status can be sent to UDC gate array and hence the CPU.

READY- (pin 68) is inverted by an open-collector inverter and connected to the RDY input of the 65C02 (pin 2) and pin 21 of the peripheral expansion connector. When the CPU attempts to read the contents of the data register in the UDC gate array through the I/O location $CFF8, READY- will go high so that the current CPU cycle will be stretched. This continues until a valid disk data is present in the UDC data register so that READY- will return low and the CPU can complete the read cycle.

RAM- (pin 1) is connected to pin 20 (OE-) and pin 18 (CE-) of the 6116 static RAM. This extra RAM is used by the CPU as temporary storage of data and parameters in a 3.5" drive data transfer transaction. Whenever the CPU wants to access the static RAM, UDC decodes the system address bus and asserts RAM- (low) to enable the RAM chip.

RAMA10 (pin 2) is connected to pin 19 (A10) of 6116. It is used for selecting one of two 1 K-byte banks in the RAM chip for read/write.

11.10 Expansion RAM control

The expansion RAM is controlled by the MMU and VDG gate arrays. All the control signals are routed to the optional expansion RAM card through the connecting pins mounted on the main board PCB.

Signals ERA0 to ERA8 generated by the UDC gate array (pin 30 to 38) are buffered on the RAM board to form the expansion RAM (41256) address lines.

Signal RASB- (pin 33) generated by the MMU gate array is buffered on the RAM board and connected to the row address strobe input (RAS-) of the dynamic RAM chips. At the falling-edge of RASB-, the row address on ERA0 to ERA8 is strobed into the RAM chips.

AX generated by the MMU gate array (pin 31) is connected to pin 71 of UDC. This is the dynamic RAM address multiplex signal. When it is high, row address is selected. Otherwise, column address will be selected to output on ERA0 to ERA8.

Signal ECAS0- from MMU (pin 30) is connected to pin 74 of UDC. It is de-multiplexed in UDC to generate the four signals ECAS1- to ECAS4- which are buffered on the expansion RAM card to become the column address strobe inputs for the four rows of RAM chips. Only one of the four signals can be low at a time to strobe the column address into the associated row of RAM chips.

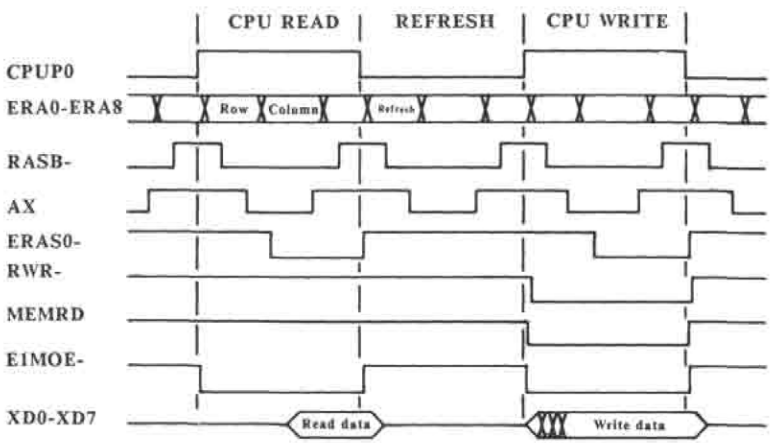
RWR- from MMU (pin 17) is buffered on the RAM card and connected to the write-enable inputs of the RAM chips. If it is low, a RAM write cycle will be performed. Otherwise, a read cycle will be performed on the RAM chips.

The expansion RAM data bus is buffered by 74HCT245 on the RAM card before connecting to the system data bus ("LASER 128") OR CPU data bus ("LASER 128 EX").

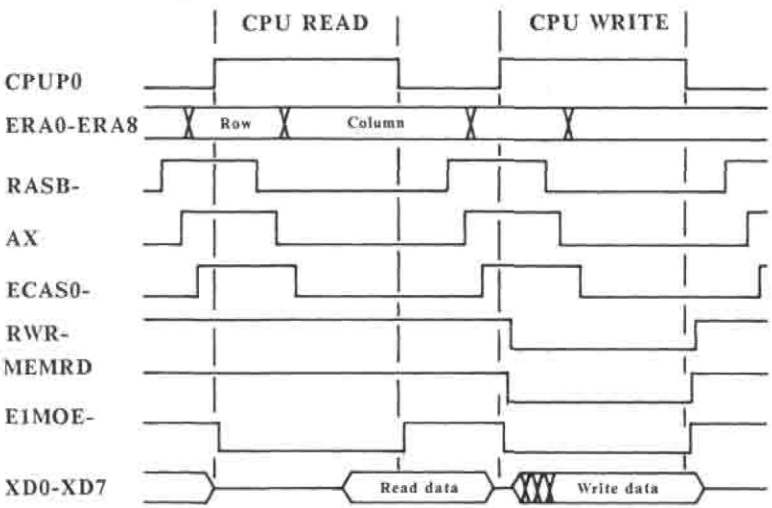
The direction control input of the expansion RAM data bus transceiver is connected to MEMRD (pin 18 of MMU). When MEMRD is low, data is transferred from the system or CPU data bus to the expansion RAM data bus. When MEMRD is high, the direction of data transfer is reversed.

The output buffer control input of the expansion RAM data bus transceiver is connected to E1MOE- (pin 28 of MMU). It will go low whenever the location $C0D3 is accessed after the expansion RAM is enabled by accessing any locations in the range $C500 to $C5FF while the "INT/EXT port 5" switch is set to the "INT PORT 5" position.

Fig. 11-19 shows the timing of the expansion RAM under different CPU clock speed.



(a) 1 MHz



(b) 2.3 MHz, 3.6 MHz

Fig. 11-19 Timing of expansion RAM

The signal X1MREF generated by the MMU gate array (pin 29) is connected to the UDC gate array (pin 75). When this signal is high, the outputs of the expansion RAM refresh counter in the UDC gate array will be driven into ERA0 to ERA8. At the falling-edge of RASB-, the contents of the addressed row of memory cells in the dynamic RAM chips will be refreshed. When X1MREF is low, normal read/write operations will be performed on the expansion RAM.

11.11 Serial Port Control

The data transfer operations on the two serial ports of the computer are controlled by the MMU gate array, UDC gate array, two 6551 ACIAs and 1488/1489 level shifters.

The signal ACIACS1- from the MMU gate array (pin 60) is connected to one of the chip-select inputs (pin 3) of the ACIAs. When either the locations from $C0A8 to $C0AB is accessed or the locations from $C098 to $C09B is accessed while the "PARALLEL/SERIAL" switch is set to the "SERIAL" position, ACIACS1- will go low to enable one of the two ACIAs.

The system address lines BA0 and BA1 are connected to the register select inputs of the ACIAs to select one of the internal registers for read/write. Data transfer operations between the CPU and the ACIAs are synchronized to the clock signal CPUP0 which is connected to pin 27 of the ACIAs. The signal MEMRD (pin 18 of MMU) is connected to the read/write input (pin 28) of the ACIAs. When MEMRD is high, read operation will be performed to transfer the contents of the addressed ACIA register to the system data bus. Otherwise, a register write is performed to transfer the contents of the system data bus to the addressed ACIA register.

In NTSC versions, the reference clock for the internal baud rate generator of the ACIAs (pin 6) are generated by the UDC gate array. The signal S7M from the MMU gate array (pin 66) is routed to the UDC gate array (pin 64) where it is divided by four to generate the signal ACIACK (pin 59) having a frequency of 1.79 MHz.

For PAL versions, there is a separate crystal oscillator operating at a frequency of 3.686 MHz. The oscillator output is divided by two to generate the signal ACIACK.

The transmit data (pin 10) request-to-send (pin 8) outputs of the two ACIA's are converted to RS232C levels (+3V to +12V for logic 0, -3V to -12V for logic 1) by 1488 before connecting to the serial interface connectors on the back panel of the computer.

The receive data and data-set-ready signals on the serial interface connectors are converted to TTL levels by 1489 before connecting to the receive data (pin 12) and data-carrier-detect (pin 16) inputs of the ACIAs.

The IRQ- outputs (pin 26) of the two ACIAs are wired together and connected to the IRQ- input (pin 4) of 65C02. Through this interrupt request line, the ACIAs can inform the CPU that certain events have occurred, e.g. transmit data register empty, receive data register full, data parity error, data overrun error and so on.

For a more detailed description on the operation of the ACIA, please refer to the data sheets published by the manufacturers.

11.12 Parallel Printer Control

The task of parallel printer interfacing is handled by the MMU gate array and a printer data latch (74HCT374).

When the CPU writes to the locations $C090 to $C09F while the "PARALLEL/SERIAL" switch is set to the "PARALLEL" position, an active-low pulse will be generated at the output PTRLTH of the MMU gate array (pin 57) which is connected to the clock input of the data latch (pin 11).

At the rising-edge of PTRLTH, the contents in the system data bus will be clocked into the printer data latch. The data latch outputs PD0 to PD7 are connected to pin 1 to pin 8 of the parallel printer connector.

At about 0.5 us after the printer data is latched, the output PTRSTB- of the MMU gate array (pin 58), which is connected to pin 10 of the parallel printer connector, will generate an active-low pulse of 1 us in duration to inform the printer that valid data is available on the printer data lines PD0 to PD7. The printer will then strobe in the printer data from the data lines.

The printer will also assert the active-high signal PTRBSY which is available at pin 9 of the parallel printer connector to inform the host computer that the printer is busy printing the previous data and therefore cannot accept any more data at this moment. This signal is "clamped" to ground and +5V before connecting to pin 56 of the MMU gate array. When the CPU reads the location $C1C1, this signal will be propagated to bit 7 of the system data bus through the MMU gate array.

Before writing to the parallel printer, the CPU should first check the printer-busy status. If the printer is busy, the CPU should wait until the printer has finished printing the previous data and de-asserts PTRBSY before sending another data to the printer.

11.13 Joystick and paddle control

The joysticks and paddles connected to the game connector on the back panel of the computer generates two types of signals, namely switch and analog inputs.

The switch inputs are connected to pin 1 (GAMESW1) and pin 7 (GAMESW0) of the game connector. These two pins are pulled down in the joystick or paddle by a resistor having a resistance between 220 Ohm and 470 Ohm. When the game switches are closed, these pins will be connected to +5V. GAMESW0 and GAMESW1 are isolated by opto-couplers 4N27 before connecting to pin 46 (GI1) and pin 47 (GI2) of the UDC gate array respectively. When the CPU reads the locations $C061 and $C062, these two signals will be driven into bit 7 of the system data bus through pin 76 of UDC.

The opto-coupler outputs for GAMESW0 and GAMESW1 are also connected to the "open-triangle" and "close-triangle" keys on the keyboard respectively. As a result, these two keys are functionally equivalent to the two game switches.

The two analog inputs are connected to pin 5 (PDL0) and pin 8 (PDL1) of the game connector. These two pin are pulled up to +5V in the joystick or paddle by 150 K-ohm variable resistors.

The analog inputs are connected to the inputs of a timer circuit in the host computer which consists of a 556 dual-timer and some RC circuits. The timer outputs for PDL0 and PDL1 are connected to pin 49 (GI4) and pin 50 (GI5) of the UDC gate array respectively.

When the CPU reads the locations $C064 and $C065, the states of the timer outputs for PDL0 and PDL1 will be transferred to bit 7 of the system data bus.

The trigger inputs of 556 (pin 6 and 8) are connected to the output C07X- from the UDC gate array (pin 53). Whenever the CPU accesses the locations from $C070 to $C07F, this signal will be driven low to start the timer. The timer outputs will then go high immediately.

Depending on the position of the rotary knob on the paddle or the joystick and hence the setting of the variable resistor, the time constant of the RC circuits will be different and hence the timer outputs will return to their original state (low) after a variable amount of time. By using a simple polling loop to read the timer outputs, it is possible to determine the position of the joystick or rotary knob on the paddle from the duration of the positive pulse generated at the timer outputs.

11.14 Mouse control

The game connector can also accepts a mouse as an input device. To distinguish it from a joystick or paddle, pin 1 of the game connector is grounded in the mouse.

The control voltage for the 556 dual-timer is derived from one of the opto-couplers, the input of which is connected to pin 1 of the game connector. When it is grounded, the opto-coupler is turned on, making the control voltage for the timers close to +5V and thus disabling the timers.

Pin 7 of the game connector is connected to the button on the mouse. If the mouse button is pressed, this pin will be grounded. Otherwise, this pin will be open.

As described in the previous section, pin 7 of the game connector is connected to the input of an opto-coupler, the output of which is wired together with the "open-triangle" key and connected to the UDC gate array.

To distinguish the mouse button from the "open-triangle" key, pin 7 of the game connector is connected to the input of another opto-coupler, the output of which is connected to pin 48 (GI3) of the UDC gate array.

If the mouse button is pressed, the opto-coupler will be turned on so that its output will be low. Otherwise, the output of the opto-coupler will be high.

When the CPU reads from the location $C063, the state of the opto-coupler output will be transferred to bit 7 of the system data bus through the UDC gate array and strobed in by the CPU. By sensing the state of this signal, the CPU can tell whether the mouse button or the "open-triangle" key is pressed.

Pin 4 (XDIR) and pin 5 (XINT) of the game connector is connected to pin 51 of UDC and pin 54 of MMU respectively. Whenever the mouse is moved on a flat surface horizontally, a series of pulses will be generated at these two pins.

Depending on the internal register settings, the MMU may generate an interrupt request to the CPU via the IRQ- (pin 68) output at either the rising-edge or falling-edge of the input XINT.

When the CPU reads from the location $C066, the state of the signal XDIR will be transferred to bit 7 of the system data bus through the UDC gate array. By sensing the state of this signal, the CPU can determine the direction of horizontal movement.

Similarly, pin 8 (YDIR) and pin 9 (YINT) of the game connector is connected to pin 52 of UDC and pin 55 of MMU respectively.

Any vertical motion of the mouse will generate a series of pulses at these two pins. Either the rising-edge or falling-edge of the signal YINT can generate an interrupt to the CPU via the MMU gate array output IRQ-.

Reading from the location $C067 causes the state of YDIR to be transferred to bit 7 of the system data bus. By sensing the state of this signal, the CPU can determine the direction of vertical motion.

The VDG gate array also outputs a signal VBLK- (pin 83) which is connected to pin 53 of the MMU gate array. This signal is low during the vertical blanking interval and high during the active display area. The timing is shown in Fig. 11-20.

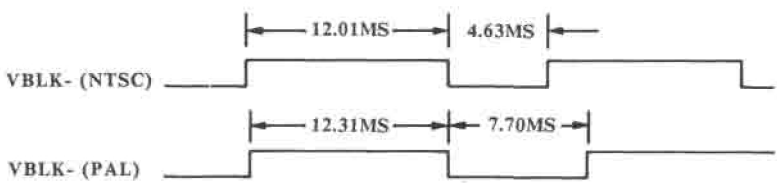


Fig. 11-20 Timing of vertical blanking

At the falling-edge of this signal, i.e. the beginning of the vertical blanking interval, an interrupt request to the CPU may be generated by the MMU gate array. Using this feature, the program may first determine the direction of mouse movement and then update the screen accordingly during the vertical blanking interval so as to produce a flicker-free graphics display.

11.15 Power supply

The power supply for the computer is made up of two parts, namely the external AC power adaptor and the on-board DC voltage converter.

The AC power adaptor consists of a step-down voltage transformer and a rectifier. It converts the AC power line voltage into a DC voltage which is fed into the computer through the power connector on on the back panel. The specifications of the AC adaptor are listed in Table 11-1.

Input AC voltage	110V/60Hz or 220V/50Hz
Output DC voltage	17.0V
Maximum tolerance	+/- 5%
Maximum current	1.8A

Table 11-1 AC power adaptor electrical specifications

The DC voltage generated by the AC power adaptor is in turn regulated by the DC voltage converter on the motherboard to provide the +5V, +12V and -12V power supplies.

The "heart" of the +5V regulator is the TL494 (U2) which is a fixed frequency, pulse width modulation control circuit for switch-mode power supply control. It has an internal linear sawtooth oscillator which is set to operate at a frequency of about 28 KHz by two external components, namely C13 and R23, connected to pin 5 and pin 6 respectively.

The sawtooth voltage generated at pin 5 is compared with the outputs of two error amplifiers inside the chip. Whenever the sawtooth voltage is greater than the outputs of the amplifiers, the two internal switching transistors will be turned on. Thus a switching pulse having the same frequency as the sawtooth oscillator is generated at the collectors of the switching transistors.

The emitters of the two transistors (pin 9 and pin 10) are grounded while the collectors (pin 8 and pin 11) are tied together and connected to the base of the PNP power transistor Q2 through the resistor R15. The emitter of the power transistor is connected to the filtered DC 17V input. Whenever the two switching transistors are turned on, the power transistor will also be turned on, transferring power from the emitter to the collector. The collector output of the power transistor is then stepped down by a "buck" circuit and filtered to obtain the +5V power supply.

By varying the error amplifier input control signals of TL494, the duty cycle of the switching pulse output and hence the average power delivered to the collector of the power transistor and the output voltage can be controlled.

TL494 has an internal voltage regulator which generates a +5V reference voltage at pin 14, with an accuracy of +/-5%. This is attenuated by the potential divider network made up of resistors R34 and R107 and connected to the inverting input of comparator 1 (pin 2) of TL494.

On the other hand, the +5V output is attenuated by another potential divider network made up of resistors R27, VR2 and R28 before connecting to the non-inverting input of comparator 1 (pin 1).

The attenuated output voltage is compared with the attenuated reference voltage. If the former is larger than the latter, then TL494 will generate a switching pulse with a smaller duty cycle so that the amount of power delivered to the collector of the power transistor and hence the output voltage will be decreased. The opposite is also true. This negative feedback operation will continue until the two error amplifier inputs are equal so that a stable and accurate +5V power supply is generated.

By adjusting the variable resistor VR2, the ratio of the potential divider at the non-inverting input of comparator 1 and hence the +5V output voltage can be varied slightly. This is necessary due to the variation of component values and the tolerance of the reference voltage. The variable resistor VR2 is adjusted in the factory to produce an output voltage of +5V +/-5%.

The resistor R29 is connected in series between the collector of the power transistor Q2 and the +5V output. When the supply is loaded, current flows through this resistor, thus producing a voltage drop directly proportional to the load current.

The +5V output is connected to the inverting input of error amplifier 2 of TL494 (pin 15) through resistor R26. The voltage at the other end of resistor R29 is attenuated by the potential divider made up of R24 and R25 before connecting to the non-inverting input of error amplifier 2 of TL494 (pin 16).

When the supply is overloaded, the voltage drop across resistor R29 is so large that the voltage at the non-inverting input of error amplifier 2 is greater than that of the inverting input. This causes the switching pulse and hence the power transistor to be turned off momentarily so that the load current will be cut off and the voltage drop across the current-limiting resistor R29 will decrease until it falls below the cross-over point and the switching pulse is turned on again. This feature provides overload and short circuit protection to the power supply.

The collector of the power transistor Q2 is connected to the primary coil of the power transformer T2 before connecting to the current-limiting resistor. This coil serves two purposes.

First of all, it forms part of a "buck" circuit to step down the collector voltage to the +5V output voltage. Secondly, it couples part of the electrical energy from the primary coil to the secondary coil of the transformer. The secondary coil forms part of a "boost variation" circuit from which the -12V power supply is derived.

The filtered +17 DC input is also connected to a conventional voltage regulator made up of transistors and resistors to generate the +12V output.

The electrical specifications of the DC voltage converter is shown in Table 11-2

		Minimum	Typical	Maximum	Unit
Input voltage		+13.0	+17.0	+18.7	DC V
Full load current	+12V	-	1	-	A
	+5	-	1.5	-	A
	-12	-	100	-	mA
Output ripple	+12	-	-	100	mV p-p
	+5	-	-	150	mV p-p
	-12	-	-	100	mV p-p
Current limit	+12	2.0	2.5	3.0	A
	+5	2.2	2.4	2.6	A
	-12	162	180	198	mA
Short circuit current	+12	16	20	24	mA
	+5	80	100	120	mA
	-12	162	180	198	mA

Table 11-2 DC voltage converter electrical specifications

APPENDIX A
CPU SPEED CONTROL IN "LASER 128 EX"

A. CPU speed control in "LASER 128 EX"

With its high-speed central processor (4 MHz 65C02), the "LASER 128 EX" is capable of running programs at up to 3.6 times its normal speed, a feature which is not shared by the "LASER 128".

Note: The speed-control technical information provided below should be used only for reference purposes. System speed is intended to be controlled by the user (by holding down the 1, 2, or 3 key while turning on the computer or pressing CTRL-RESET). If an application program arbitrarily changes system speed without regard to user preferences, this unfortunately takes the desired control away from the user.

The clock frequency of the CPU is software-selectable to be 1 MHz, 2.3 MHz or 3.6 MHz through the write-only I/O location $C074 which is usually referred to as the speed register. Only bit 5 to bit 7 of the speed register are used.

Bit 6 and bit 7 of the speed register are used for selecting the highest operating frequency of the CPU. Table A-1 shows the CPU speed for various speed register bit settings.

Speed register ($C074)		Maximum CPU operating frequency
Bit 7	Bit 6	
0	0	1 MHz
0	1	1 MHz
1	0	2.3 MHz
1	1	3.6 MHz

Table A-1 Selection of CPU speed

Notice that the CPU clock frequencies specified in the table are only maximum values. The actual operating frequency of the CPU at a particular instant and hence the average CPU clock speed may be smaller. There are a couple of reasons for this.

First of all, some memory chips and I/O devices may not be able to operate at the "turbo" speed due to some timing restrictions. For example, the program ROM (27256) and expansion RAM can only operate at either 1 MHz or 2.3 MHz. If the maximum CPU speed is chosen to be 3.6 MHz, then the CPU clock will automatically slow down to 2.3 MHz when these devices are being accessed.

The CPU clock will also be slowed down to 1 MHz momentarily under the following conditions:

- Accessing the ACIA chips, i.e. addresses in the ranges $C090 to $C09F and $C0A0 to $C0AF.

- Accessing the address ranges $C0D0 to $C0DF and $C500 to $C5FF while external port 5 is selected using the "INT/EXT PORT 5" switch.

- Accessing the address ranges $C0F0 to $C0FF and $C700 to $C7FF while external port 7 is selected using the "INT/EXT PORT 7" switch.

- Accessing the address range $C800 to $CFFF while external port 5 or external port 7 is selected.

- Pin 32 of the peripheral expansion connector (INH-) is pulled low by an external interface card plugged into the optional expansion box.

Secondly, for some time-critical application programs, it is necessary to slow down the CPU clock to 1 MHz in order to maintain compatibility with "LASER 128" so that the program can run correctly. Moreover, the CPU clock frequency is not constant in the two "turbo" modes and hence cannot generate precise software timing loop.

This is particularly important in disk drive interfacing since the data transfer operations between the disk drive and the main memory of the computer are directly handled by the CPU. As a result, the CPU clock will be slowed down to 1 MHz under the following conditions:

- The disk drives attached to the computer (either built-in or through the external drive connector) are enabled, i.e. when the "ENABLE" software-switch is turned on by accessing the I/O location $C0E9.

- The 5.25" disk drive firmware resided in $C600 to $C6FF and $C800 to $CFFF (bank-switched ROM space) is being executed.

For an external disk drive controller card inserted in the expansion box (external port 7) to work correctly, it is also necessary for the CPU clock to slow down to 1 MHz under the following conditions:

- The external disk drive is enabled by accessing the location $C0F9.

- The external disk drive firmware located at $C700 to $C7FF and $C800 to $CFFF (bank-switched ROM) is being executed.

However, some interface cards are not that timing-critical and may be able to operate at the superior "turbo" speeds. As a result, the "external port 7 drive detect" feature mentioned above is optional and can be selected in software through bit 5 of the speed register.

If bit 5 of the speed register is set to one, then the CPU will slow down to 1 MHz when the external disk controller is activated as described above. If it is set to zero, then the CPU will not be slowed down except when the addresses $C0F0 to $C0FF and $C700 to $C7FF are being accessed.

This bit of the speed register can only be set once after power-up by writing to the location $C074. Writing to this bit again has no effect on the setting.

APPENDIX B
KEYBOARD LAYOUTS AND KEY CODES

Fig B-1 and Fig B-2 are the keyboard layouts of the U.S.A. standard (Sholes) and the U.S.A. simplified (Dvorak) respectively.

Table B-1 and Table B-2 show the corresponding ASCII codes of the Sholes and the Dvorak keyboard layout.

In Table B-2, the column KEY refers to alphanumeric letters on the keytops as in Fig B-1.

In some versions of the Laser family, the ALT/STD switch is not provided. The U.S.A. standard (Sholes) is the default keyboard layout.

Fig B-1 USA standard keyboard layout

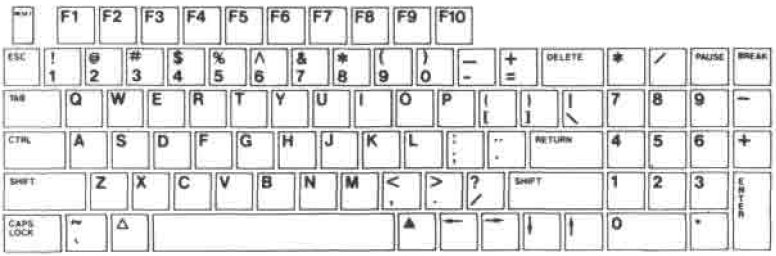


Table B-1 The ASCII code of standard USA keyboard.

KEY	NORM	CHAR	CTRL	CHAR	SHIFT	CHAR	BOTH	CHAR
F1	00	NUL	00	NUL	00	NUL	00	NUL
F2	01	SOH	01	SOH	01	SOH	01	SOH
F3	02	STX	02	STX	02	STX	02	STX
F4	03	ETX	03	ETX	03	ETX	03	ETX
F5	04	EOT	04	EOT	04	EOT	04	EOT
F6	05	ENQ	05	ENQ	05	ENQ	05	ENQ
F7	06	ACK	06	ACK	06	ACK	06	ACK
F8	07	BEL	07	BEL	07	BEL	07	BEL
F9	0C	FF	0C	FF	0C	FF	0C	FF
F10	18	CAN	18	CAN	18	CAN	18	CAN
0)	30	0	30	0	29	)	29	)
1 !	31	1	31	1	21	!	21	!
2 @	32	2	00	NUL	40	@	00	NUL
3 #	33	3	33	3	23	#	23	#
4 $	34	4	34	4	24	$	24	$
5 %	35	5	35	5	25	%	25	%
6 ^	36	6	1E	RS	5E	^	1E	RS
7 &	37	7	37	7	26	&	26	&
8 *	38	8	38	8	2A	*	2A	*
9 (	39	9	39	9	28	(	28	(
A	61	a	01	SOH	41	A	01	SOH
B	62	b	02	STX	42	B	02	STX
C	63	c	03	ETX	43	C	03	ETX
D	64	d	04	EOT	44	D	04	EOT
E	65	e	05	ENQ	45	E	05	ENQ
F	66	f	06	ACK	46	F	06	ACK
G	67	g	07	BEL	47	G	07	BEL
H	68	h	08	BS	48	H	08	BS
I	69	i	09	HT	49	I	09	HT
J	6A	j	0A	LF	4A	J	0A	LF
K	6B	k	0B	VT	4B	K	0B	VT
L	6C	l	0C	FF	4C	L	0C	FF
M	6D	m	0D	CR	4D	M	0D	CR
N	6E	n	0E	SO	4E	N	0E	SO

KEY	NORM	CHAR	CTRL	CHAR	SHIFT	CHAR	BOTH	CHAR
O	6F	o	0F	SI	4F	O	0F	SI
P	70	p	10	DLE	50	P	10	DLE
Q	71	q	11	DC1	51	Q	11	DC1
R	72	r	12	DC2	52	R	12	DC2
S	73	s	13	DC3	53	S	13	DC3
T	74	t	14	DC4	54	T	14	DC4
U	75	u	15	NAK	55	U	15	NAK
V	76	v	16	SYN	56	V	16	SYN
W	77	w	17	ETB	57	W	17	ETB
X	78	x	18	CAN	58	X	18	CAN
Y	79	y	19	EM	59	Y	19	EM
Z	7A	z	1A	SUB	5A	Z	1A	SUB
- _	2D	-	1F	US	5F	_	1F	US
= +	3D	=	3D	=	2B	+	2B	+
[{	5B	[	1B	ESC	7B	{	1B	ESC
] }	5D	]	1D	GS	7D	}	1D	GS
\ |	5C	\	1C	FS	7C	|	1C	FS
; :	3B	;	3B	;	3A	:	3A	:
' "	27	'	27	'	22	"	22	"
, <	2C	,	2C	,	3C	<	3C	<
. >	2E	.	2E	.	3E	>	3E	>
/ ?	2F	/	2F	/	3F	?	3F	?
` ~	60	`	60	`	7E	~	7E	~
SPACE	20	SP	20	SP	20	SP	20	SP
←	08	BS	08	BS	08	BS	08	BS
→	15	NAK	15	NAK	15	NAK	15	NAK
↓	0A	LF	0A	LF	0A	LF	0A	LF
↑	0B	VT	0B	VT	0B	VT	0B	VT
0	30	0	30	0	30	0	30	0
1	31	1	31	1	31	1	31	1
2	32	2	32	2	32	2	32	2
3	33	3	33	3	33	3	33	3
4	34	4	34	4	34	4	34	4
5	35	5	35	5	35	5	35	5
6	36	6	36	6	36	6	36	6
7	37	7	37	7	37	7	37	7
8	38	8	38	8	38	8	38	8
9	39	9	39	9	39	9	39	9
*	2A	*	2A	*	2A	*	2A	*
/	2F	/	2F	/	2F	/	2F	/
-	2D	-	2D	-	2D	-	2D	-
+	2B	+	2B	+	2B	+	2B	+
.	2E	.	2E	.	2E	.	2E	.
ESC	1B	ESC	1B	ESC	1B	ESC	1B	ESC
TAB	09	HT	09	HT	09	HT	09	HT
DELETE	7F	DEL	7F	DEL	7F	DEL	7F	DEL
RETURN	0D	CR	0D	CR	0D	CR	0D	CR
PAUSE	13	DC3	13	DC3	13	DC3	13	DC3
BREAK	03	ETX	03	ETX	03	ETX	03	ETX
ENTER	0D	CR	0D	CR	0D	CR	0D	CR

Fig B-2 Simplified (DVORAK) Keyboard layout

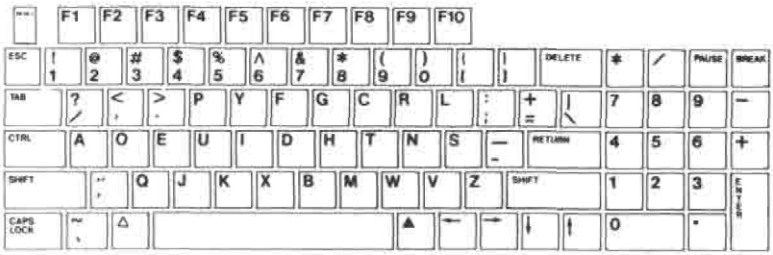


Table B-2 The ASCII code of simplified (DVORAK) keyboard

KEY	NORM	CHAR	CTRL	CHAR	SHIFT	CHAR	BOTH	CHAR
F1	00	NUL	00	NUL	00	NUL	00	NUL
F2	01	SOH	01	SOH	01	SOH	01	SOH
F3	02	STX	02	STX	02	STX	02	STX
F4	03	ETX	03	ETX	03	ETX	03	ETX
F5	04	EOT	04	EOT	04	EOT	04	EOT
F6	05	ENQ	05	ENQ	05	ENQ	05	ENQ
F7	06	ACK	06	ACK	06	ACK	06	ACK
F8	07	BEL	07	BEL	07	BEL	07	BEL
F9	0C	FF	0C	FF	0C	FF	0C	FF
F10	18	CAN	18	CAN	18	CAN	18	CAN
0)	30	0	30	0	29	)	29	)
1 !	31	1	31	1	21	!	21	!
2 @	32	2	00	NUL	40	@	00	NUL
3 #	33	3	33	3	23	#	23	#
4 $	34	4	34	4	24	$	24	$
5 %	35	5	35	5	25	%	25	%
6 ^	36	6	1E	RS	5E	^	1E	RS
7 &	37	7	37	7	26	&	26	&
8 *	38	8	38	8	2A	*	2A	*
9 (	39	9	39	9	28	(	28	(
A	61	a	01	SOH	41	A	01	SOH
B	78	x	18	CAN	58	X	18	CAN
C	6A	j	0A	LF	4A	J	0A	LF
D	65	e	05	ENQ	45	E	05	ENQ
E	2E	.	2E	.	3E	>	3E	>
F	75	u	15	NAK	55	U	15	NAK
G	69	i	09	HT	49	I	09	HT
H	64	d	04	EOT	44	D	04	EOT
I	63	c	03	ETX	43	C	03	ETX
J	68	h	08	BS	48	H	08	BS
K	74	t	14	DC4	54	T	14	DC4
L	6E	n	0E	SO	4E	N	0E	SO
M	6D	m	0D	CR	4D	M	0D	CR
N	62	b	02	STX	42	B	02	STX
O	72	r	12	DC2	52	R	12	DC2

KEY	NORM	CHAR	CTRL	CHAR	SHIFT	CHAR	BOTH	CHAR
P	6C	l	0C	FF	4C	L	0C	FF
Q	2F	/	2F	/	3F	?	3F	?
R	70	p	10	DLE	50	P	10	DLE
S	6F	o	0F	SI	4F	O	0F	SI
T	79	y	19	EM	59	Y	19	EM
U	67	g	07	BEL	47	G	07	BEL
V	6B	k	0B	VT	4B	K	0B	VT
W	2C	,	2C	,	3C	<	3C	<
X	71	q	11	DC1	51	Q	11	DC1
Y	66	f	06	ACK	46	F	06	ACK
Z	27	'	27	'	22	"	22	"
- _	5B	[	1B	ESC	7B	{	1B	ESC
= +	5D	]	1D	GS	7D	}	1D	GS
[{	3B	;	3B	;	3A	:	3A	:
] }	3D	=	3D	=	2B	+	2B	+
\ |	5C	\	1C	FS	7C	|	1C	FS
; :	73	s	13	DC3	53	S	13	DC3
' "	2D	-	1F	US	5F	-	1F	US
, <	77	w	17	ETB	57	W	17	ETB
. >	76	v	16	SYN	56	V	16	SYN
/ ?	7A	z	1A	SUB	5A	Z	1A	SUB
` ~	60	`	60	`	7E	~	7E	~
SPACE	20	SP	20	SP	20	SP	20	SP
←	08	BS	08	BS	08	BS	08	BS
→	15	NAK	15	NAK	15	NAK	15	NAK
↓	0A	LF	0A	LF	0A	LF	0A	LF
↑	0B	VT	0B	VT	0B	VT	0B	VT
0	30	0	30	0	30	0	30	0
1	31	1	31	1	31	1	31	1
2	32	2	32	2	32	2	32	2
3	33	3	33	3	33	3	33	3
4	34	4	34	4	34	4	34	4
5	35	5	35	5	35	5	35	5
6	36	6	36	6	36	6	36	6
7	37	7	37	7	37	7	37	7
8	38	8	38	8	38	8	38	8
9	39	9	39	9	39	9	39	9
*	2A	*	2A	*	2A	*	2A	*2A
/	2F	/	2F	/	2F	/	2F	/
-	2D	-	2D	-	2D	-	2D	-
+	2B	+	2B	+	2B	+	2B	+
.	2E	.	2E	.	2E	.	2E	.
ESC	1B	ESC	1B	ESC	1B	ESC	1B	ESC
TAB	09	HT	09	HT	09	HT	09	HT
DELETE	7F	DEL	7F	DEL	7F	DEL	7F	DEL
RETURN	0D	CR	0D	CR	0D	CR	0D	CR
PAUSE	13	DC3	13	DC3	13	DC3	13	DC3
BREAK	03	ETX	03	ETX	03	ETX	03	ETX
ENTER	0D	CR	0D	CR	0D	CR	0D	CR

APPENDIX C
65C02 PROGRAMMING SPECIFICATION (The followings are extracted from GTE G65SCXXX data sheets)

Addressing Modes

Fifteen addressing modes are available to the user of the GTE G65SCXXX family of microprocessors. The addressing modes are described in the following paragraphs:

Implied Addressing

In the implied addressing mode, the address containing the operand is implicitly stated in the operation code of the instruction.

Accumulator Addressing

This form of addressing is represented with a one byte instruction and implies an operation on the accumulator.

Immediate Addressing

With immediate addressing, the operand is contained in the second byte of the instruction; no further memory addressing is required.

Absolute Addressing

For absolute addressing, the second byte of the instruction specifies the eight low order bit of the effective address while the third byte specifies the eight high order bits. Therefore, this addressing mode allows access to the total 65K bytes of addressable memory.

Zero page Addressing

Zero page addressing allows shorter code and execution times by only fetching the second byte of the instruction and assuming a zero high address byte. The careful use of zero page addressing can result in significant increase in code efficiency.

Absolute Indexed Addressing

Absolute indexed addressing is used in conjunction with X and Y index register and is referred to as "Absolute, X", and "Absolute, Y". The effective address is formed by adding the contents of X and Y to the address contained in the second and third bytes of the instruction. This mode allows the index register to contain the index or count value and the instruction to contain the base address. This type of indexing allows any location referencing and the index to modify multiple fields resulting in reduced coding and execution time.

®Ackn: GTE Microcircuits

Zero Page Indexed Addressing

Zero page absolute addressing is used in conjunction with the index register and is referred to as "Zero Page, X" or "Zero Page, Y." The effective address is calculated by adding the second byte to the contents of the index register. Since this is a form of "Zero Page" addressing, the content of the second byte references a location in page zero. Additionally, due to the "Zero Page" addressing nature of this mode, no carry is added to the high order eight bits of memory and crossing of page boundaries does not occur.

Relative Addressing

Relative addressing is used only with branch instruction; it establishes a destination for the conditional branch.

Zero Page Indexed Indirect Addressing

With zero page indexed indirect addressing (usually referred to as Indirect X) the second byte of the instruction is added to the contents of the X index register; the carry is discarded, the result of this addition points to a memory location on page zero whose contents is the low order eight bits of the effective address. The next memory location in page zero contains the high order eight bit of the effective address. Both memory locations specifying the high and low order bytes of the effective address must be in page zero.

Absolute Indexed Indirect Addressing (Jump Instruction Only)

With absolute indexed indirect addressing, the contents of the second and third instruction bytes are added to the X register. The result of this addition points to a memory location containing the lower-order eight bits of the effective address. The next memory location contains the high-order eight bits of the effective address.

Indirect Indexed Addressing

This form of addressing is usually referred to as Indirect, Y. The second byte of the instruction points to a memory location in page zero. The contents of this memory location is added to the contents of the Y index register, the result being the low order eight bits of the effective address. The carry from this addition is added to the contents of the next page zero memory location, the result being the high order eight bits of the effective address.

Zero Page Indirect Addressing

In this form of addressing, the second byte of the instruction contains the low order eight bits of a memory location. The high order eight bits is always zero. The contents of the fully specified memory location is the low order byte of the effective address. The next memory location contains the high order byte of the effective address.

Absolute Indirect Addressing (Jump Instruction Only)

The second byte of the instruction contains the low order eight bits of a memory location. The high order eight bits of that memory location is contained in the third byte of the instruction. The contents of the fully specified memory location is the low order byte of the effective address. The next memory location contains the high order byte of the effective address which is loaded into the 16 bits of the program counter.

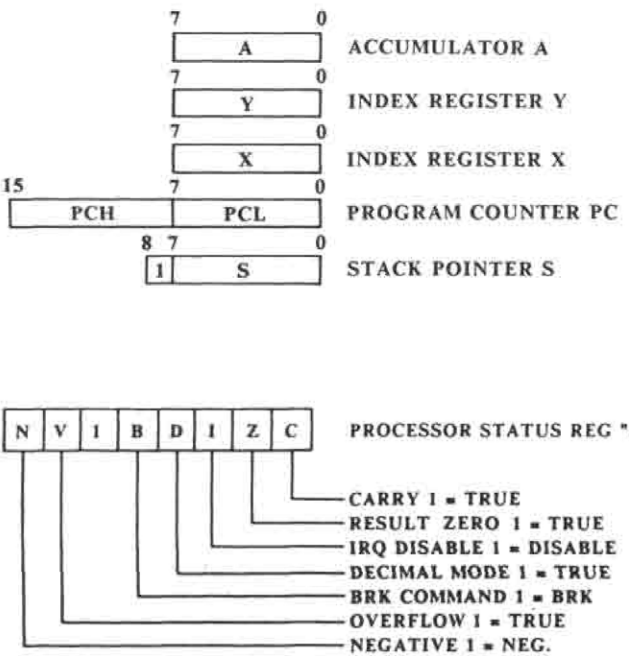


Fig. C-1. Microprocessor Programming Model

Table C-1 Instruction Set-Alphabetical Sequence

ADC	Add memory to Accumulator with Carry
AND	"AND" Memory with Accumulator
ASL	Shift One Bit Left
BCC	Branch on Carry Clear
BCS	Branch on Carry Set
BEQ	Branch on Result Zero
BIT	Test memory Bits with Accumulator
BMI	Branch on Result Minus
BNE	Branch on Result Not Zero
BPL	Branch on Result plus
•BRA	Branch Always
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 by One
DEX	Decrement Index X by One
DEY	Decrement Index Y by One
EOR	"Exclusive-or" Memory with Accumulator
INC	Increment 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 One Bit Right
NOP	No Operation
ORA	"OR" Memory with Accumulator
PHA	Push Accumulator on Stack
PHP	Push Processor Status on Stack
•PHX	Push Index X on Stack
•PHY	Push Index Y on Stack
PLA	Pull Accumulator from Stack
PLP	Pull Processor Status from Stack
•PLX	Pull Index X from Stack
•PLY	Pull Index Y from Stack
ROL	Rotate One Bit Left
ROR	Rotate One Bit Right
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 Bit
STA	Store Accumulator in memory
STX	Store Index X in Memory
STY	Store Index Y in Memory
•STZ	Store Zero in Memory
TAX	Transfer Accumulator to Index X
TAY	Transfer Accumulator to Index Y
•TRB	Test and Reset memory Bits with Accumulator
•TSB	Test and Set memory Bits with Accumulator
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

Note: • = New Instruction

LSD / MSD	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
0	BRK	ORA ind, X			TSB zpg	ORA zpg	ASL zpg		PHP	ORA imm	ASL A		TSB abs	ORA abs	ASL abs		0
1	BPL rel	ORA ind, Y	ORA ind		TRB zpg	ORA zpg, X	ASL zpg, X		CLC	ORA abs, Y	INC A		TRB abs	ORA abs, X	ASL abs, X		1
2	JSR abs	AND ind, X			BIT zpg	AND zpg	ROL zpg		PLP	AND imm	ROL A		BIT abs	AND abs	ROL abs		2
3	BMI rel	AND ind, Y	AND ind		BIT zpg, X	AND zpg, X	ROL zpg, X		SEC	AND abs, Y	DEC A		BIT abs, X	AND abs, X	ROL abs, X		3
4	RTI	EOR ind, X				EOR zpg	LSR zpg		PHA	EOR imm	LSR A		JMP abs	EOR abs	LSR abs		4
5	BVC rel	EOR ind, Y	EOR ind			EOR zpg, X	LSR zpg, X		CLI	EOR abs, Y	PHY			EOR abs, X	LSR abs, X		5
6	RTS	ADC ind, X			STZ zpg	ADC zpg	ROR zpg		PLA	ADC imm	ROR A		JMP ind	ADC abs	ROR abs		6
7	BVS rel	ADC ind, Y	ADC ind		STZ zpg, X	ADC zpg, X	ROR zpg, X		SEI	ADC abs, Y	PLY		JMP ind, X	ADC abs, X	ROR abs, X		7
8	BRA rel	STA ind, X			STY zpg	STA zpg	STX zpg		DEY	BIT imm	TXA		STY abs	STA abs	STX abs		8
9	BCC rel	STA ind, Y	STA ind		STY zpg, X	STA zpg, X	STX zpg, Y		TYA	STA abs, Y	TXS		STZ abs	STA abs, X	STZ abs, X		9
A	LDY imm	LDA ind, X	LDX imm		LDY zpg	LDA zpg	LDX zpg		TAY	LDA imm	TAX		LDY abs	LDA abs	LDX abs		A
B	BCS rel	LDA ind, Y	LDA ind		LDY zpg, X	LDA zpg, X	LDX zpg, Y		CLV	LDA abs, Y	TSX		LDY abs, X	LDA abs, X	LDX abs, Y		B
C	CPY imm	CMP ind, X			CPY zpg	CMP zpg	DEC zpg		INY	CMP imm	DEX		CPY abs	CMP abs	DEC abs		C
D	BNE rel	CMP ind, Y	CMP ind			CMP zpg, X	DEC zpg, X		CLD	CMP abs, Y	PHX			CMP abs, X	DEC abs, X		D
E	CPX imm	SBC ind, X			CPX zpg	SBC zpg	INC zpg		INX	SBC imm	NOP		CPX abs	SBC abs	INC abs		E
F	BEQ rel	SBC ind, Y	SBC ind			SBC zpg, X	INC zpg, X		SED	SBC abs, Y	PLX			SBC abs, X	INC abs, X		F
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	

Note: ▭ = New Op Codes

Fig. C-2 Microprocessor Op code table

MNE-MONIC	OPERATION	IMMEDIATE OP	n	#	ABSOLUTE OP	n	#	ZERO PAGE OP	n	#	(#) IMPLIED OP	n	#	(IND,X) OP	n	#	(IND),Y OP	n	#	ZPG,X OP	n	#	ABS,X OP	n	#	ABS,Y OP	n	#	RELATIVE (2) OP	n	#	INDIRECT OP	n	#	ZPG,Y OP	n	#	PROCESSOR STATUS CODE 7 6 4 3 2 1 0 N V B D I Z C	MNE-MONIC
ADC	A + M + C → A (3)	69	2	2	6D	4	3	65	3	2				61	6	2	71	5	2	75	4	2	7D	4	3	79	4	3				72	5	2				N V • • • Z C	ADC
AND	A ∧ M → A	29	2	2	2D	4	3	25	3	2				21	6	2	31	5	2	35	4	2	3D	4	3	39	4	3				32	5	2				N • • • • Z •	AND
ASL	C ← 7 … 0 ← 0				0E	6	3	06	5	2	0A	2	1							16	6	2	1E	6	3												N • • • • Z C	ASL
BCC	BRANCH IF C = 0																												90	2	2							• • • • • • •	BCC
BCS	BRANCH IF C = 1																												B0	2	2							• • • • • • •	BCS
BEQ	BRANCH IF Z = 1																												F0	2	2							• • • • • • •	BEQ
BIT	A ∧ M (5)	89	2	2	2C	4	3	24	3	2										34	4	2	3C	4	3												M M • • • Z •	BIT
BMI	BRANCH IF N = 1																												30	2	2							• • • • • • •	BMI
BNE	BRANCH IF Z = 0																												D0	2	2							• • • • • • •	BNE
BPL	BRANCH IF N = 0																												10	2	2							• • • • • • •	BPL
BRA	BRANCH ALWAYS																												80	2	2							• • • • • • •	BRA
BRK	BREAK										00	7	1																									• • 1 0 1 • •	BRK
BVC	BRANCH IF V = 0																												50	2	2							• • • • • • •	BVC
BVS	BRANCH IF V = 1																												70	2	2							• • • • • • •	BVS
CLC	0 → C										18	2	1																									• • • • • • 0	CLC
CLD	0 → D										D8	2	1																									• • • 0 • • •	CLD
CLI	0 → I										58	2	1																									• • • • 0 • •	CLI
CLV	0 → V										B8	2	1																									• 0 • • • • •	CLV
CMP	A - M	C9	2	2	CD	4	3	C5	3	2				C1	6	2	D1	5	2	D5	4	2	DD	4	3	D9	4	3				D2	5	2				N • • • • Z C	CMP
CPX	X - M	E0	2	2	EC	4	3	E4	3	2																												N • • • • Z C	CPX
CPY	Y - M	C0	2	2	CC	4	3	C4	3	2																												N • • • • Z C	CPY
DEC	DECREMENT				CE	6	3	C6	5	2	3A	2	1							D6	6	2	DE	6	3												N • • • • Z •	DEC
DEX	X - 1 → X										CA	2	1																									N • • • • Z •	DEX
DEY	Y - 1 → Y										88	2	1																									N • • • • Z •	DEY
EOR	A ⊻ M → A	49	2	2	4D	4	3	45	3	2				41	6	2	51	5	2	55	4	2	5D	4	3	59	4	3				52	5	2				N • • • • Z •	EOR
INC	INCREMENT				EE	6	3	E6	5	2	1A	2	1							F6	6	2	FE	6	3												N • • • • Z •	INC
INX	X + 1 → X										E8	2	1																									N • • • • Z •	INX
INY	Y + 1 → Y										C8	2	1																									N • • • • Z •	INY
JMP	JUMP TO NEW LOC				4C	3	3																									6C	6	3				• • • • • • •	JMP
JSR	JUMP SUB				20	6	3																															• • • • • • •	JSR
LDA	M → A	A9	2	2	AD	4	3	A5	3	2				A1	6	2	B1	5	2	B5	4	2	BD	4	3	B9	4	3				B2	5	2				N • • • • Z •	LDA
LDX	M → X	A2	2	2	AE	4	3	A6	3	2																BE	4	3							B6	4	2	N • • • • Z •	LDX
LDY	M → Y	A0	2	2	AC	4	3	A4	3	2										B4	4	2	BC	4	3													N • • • • Z •	LDY
LSR	0 → 7 … 0 → C				4E	6	3	46	5	2	4A	2	1							56	6	2	5E	6	3												0 • • • • Z C	LSR
NOP	NO OPERATION										EA	2	1																									• • • • • • •	NOP
ORA	A ∨ M → A	09	2	2	0D	4	3	05	3	2				01	6	2	11	5	2	15	4	2	1D	4	3	19	4	3				12	5	2				N • • • • Z •	ORA
PHA	A → Ms, S - 1 → S										48	3	1																									• • • • • • •	PHA
PHP	P → Ms, S - 1 → S										08	3	1																									• • • • • • •	PHP
PHX	X → Ms, S - 1 → S										DA	3	1																									• • • • • • •	PHX
PHY	Y → Ms, S - 1 → S										5A	3	1																									• • • • • • •	PHY
PLA	S + 1 → S, Ms → A										68	4	1																									N • • • • Z •	PLA
PLP	S + 1 → S, Ms → P										28	4	1																									N V 1 D I Z C	PLP
PLX	S + 1 → S, Ms → X										FA	4	1																									N • • • • Z •	PLX
PLY	S + 1 → S, Ms → Y										7A	4	1																									N • • • • Z •	PLY
ROL	C ← 7 … 0 ← C				2E	6	3	26	5	2	2A	2	1							36	6	2	3E	6	3												N • • • • Z C	ROL
ROR	C → 7 … 0 → C				6E	6	3	66	5	2	6A	2	1							76	6	2	7E	6	3												N • • • • Z C	ROR
RTI	RTRN INT										40	6	1																									N V 1 D I Z C	RTI
RTS	RTRN SUB										60	6	1																									• • • • • • •	RTS
SBC	A - M - C → A (3)	E9	2	2	ED	4	3	E5	3	2				E1	6	2	F1	5	2	F5	4	2	FD	4	3	F9	4	3				F2	5	2				N V • • • Z C	SBC
SEC	1 → C										38	2	1																									• • • • • • 1	SEC
SED	1 → D										F8	2	1																									• • • 1 • • •	SED
SEI	1 → I										78	2	1																									• • • • 1 • •	SEI
STA	A → M				8D	4	3	85	3	2				81	6	2	91	6	2	95	4	2	9D	5	3	99	5	3				92	5	2				• • • • • • •	STA
STX	X → M				8E	4	3	86	3	2																											96	4	2	• • • • • • •	STX
STY	Y → M				8C	4	3	84	3	2										94	4	2																	• • • • • • •	STY
STZ	00 → M				9C	4	3	64	3	2										74	4	2	9E	5	3													• • • • • • •	STZ
TAX	A → X										AA	2	1																									N • • • • Z •	TAX
TAY	A → Y										A8	2	1																									N • • • • Z •	TAY
TRB	A ∧ M → M (6)				1C	6	3	14	5	2																													• • • • • Z •	TRB
TSB	A ∨ M → M (6)				0C	6	3	04	5	2																													• • • • • Z •	TSB
TSX	S → X										BA	2	1																									N • • • • Z •	TSX
TXA	X → A										8A	2	1																									N • • • • Z •	TXA
TXS	X → S										9A	2	1																									• • • • • • •	TXS
TYA	Y → A										98	2	1																									N • • • • Z •	TYA

Table C-2 Operational codes, Execution Time, and Memory Requirements

Notes:

1. Add 1 to "n" if page boundary is crossed, except STA and STZ.
2. Add 1 to "n" if branch occurs to same page. Add 2 to "n" if branch occurs to different page.
3. Add 1 to "n" if decimal mode.
4. Accumulator address is included in Implied address.
5. "N" and "V" flags are unchanged in immediate mode.
6. "Z" flag indicates A^M result (Same as BIT instruction).

X	Index X
Y	Index Y
A	Accumulator
M	Memory per effective address
Ms	Memory per stack pointer
+	Add
-	Subtract
^	And
V	Or
⊻	Exclusive or
n	No. Cycles
#	No. Bytes
M6	Memory Bit #6
M7	Memory Bit #7

APPENDIX D
SERVICE INFORMATION

D.1 Trouble shoot Guide

This guide contains technical instructions to trouble shoot the Laser Computer. However, the guide is not a complete list . If the user is hesitated on certain trouble, he is advised to have the computer repaired by a qualified technician. For details of operation of individual circuitry, user should refer to the relevant chapters in this manual.

SYMPTOM

Computer System no response after power up. Power LED on top cabinet does not glow.

POSSIBLE CAUSES	SOLUTIONS
• Power cables not connected properly.	• Check the contact of the AC power plug of the transformer and the wall AC source. Check the contact of the transformer to the computer unit.
• No DC output from the transformer.	• The transformer has to be replaced by a good one.
• Open circuit of connection wires from the power switch to the main board.	• Solder the wires again properly.
• Bad connection between the keyboard and the main board.	• Check the connector and flat cable connecting the key board and the main board.
• No DC +5V or +12V.	• Hardware failure of the internal switching mode power supply. User is advised to have the computer repaired by a qualified technician.

Circuits to be checked:

a) The switching mode power supply control IC (U2).

b) The power transistor Q1, Q2 and Q3.

SYMPTOM

Power LED is on but the drive LED on the top cabinet is off.

POSSIBLE CAUSES	SOLUTIONS
• Bad connection between the keyboard and the main board.	• Check the connector and flat cable connecting the keyboard and the main board.

SYMPTOM

Power LED and Drive LED are on but the drive LED on the drive panel is off.

POSSIBLE CAUSES	SOLUTIONS
• Bad connection between the drive and the main board.	• Check the connection between the drive and the main board.
• The drive is in faulty operation.	• Replace with a good drive.

Circuits to be checked:

a) Pin 14 of J11 connector is active low in order to enable drive.

b) The +5V and +12V supply on pin 12 and 13 of J11.

SYMPTOM

The built-in drive cannot boot program properly.

POSSIBLE CAUSES	SOLUTIONS
• The diskette surface has been scratched or is dirty.	• Replace with a good diskette.
• The drive head is dirty.	• Clean the drive head with great precaution using the drive cleaning kit.
• The drive speed has been drifted.	• Use drive speed alignment software to adjust the drive speed. The drive speed adjust hole is on the bottom cabinet. The drive speed can be adjusted by using a small screw driver. The adjustment process should be done with great care.
• Internal hardware failure.	• Check gate array 3 and the drive buffer IC's.

Circuits to be checked:

a) Pin 2,4,6,8 and pin 16 of J11.

b) Buffer IC (U27) and PHA, PHB, PHC, PHD and RDDATA of gate array 3 (U38).

SYMPTOM

External drive does not work properly.

POSSIBLE CAUSES	SOLUTIONS
• Bad connection between the drive and the main computer.	• Check the connector and cable connecting the drive and the computer.
• If the external drive is 3.5 inch drive, then the cause may be due to wrong setting of slot 7 switch inside the ROM door on the bottom cabinet.	• The external slot 7 switch should be disabled.
• Internal hardware failure.	• Check gate array 3 and the drive buffer IC's.

Circuits to be checked:

a) Connector J5 for external drive.

b) The drive related signals of gate array 3 (U38) and the buffers (U13, U18 and U27).

SYMPTOM

Power LED is on but no beep sound.

POSSIBLE CAUSES	SOLUTIONS
• Bad connection between the speaker and the main board.	• Check the connection.
• The volume is set at a low level.	• Adjust the volume knob.

Circuits to be checked:

The sound output is from pin 21 of gate array 3 (U38).

SYMPTOM

Keyboard cannot function properly.

POSSIBLE CAUSES	SOLUTIONS
• Bad connection between the keyboard and the main board.	• Check the connection.
• Internal hardware failure.	• Check the keyboard controller IC (8048).

Circuits to be checked:

a) The keyboard connector (J12) connecting the main board and the keyboard.

b) The keyboard controller (U21) and the scanning decoder (U22).

SYMPTOM

No video display or display not in synchronisation.

POSSIBLE CAUSES	SOLUTIONS
• Bad connection between the monitor and the computer.	• Check the corresponding connection between the composite out or video expansion port.
• Display not in synchronization.	• Try to adjust by tuning the monitor V-hold and H-hold. • For the PAL model, if LCD is used as display the 50/60 switch inside the ROM door on the bottom cabinet should be set to 60. If monitor is used as display, the 50/60 switch should be set to 50.
• Internal hardware failure.	• Check the Gate array 2 and the video port buffer IC.

Circuits to be checked:

a) The video expansion connector (J7).

b) The gate array 2 (U25) and the buffers (U13) and related circuits (U41).

SYMPTOM

80 column text mode cannot work correctly.

POSSIBLE CAUSES	SOLUTIONS
• The 40/80 switch on the top cabinet is set to 40 when power is switched on.	• Switch to the 80 position and turn on the computer again (or cold start the computer by pressing CTRL - RESET - △.
• RAM failure.	• Check the RAM chips.
• Gate array 2 failure which is not common.	• Replace with a new one.

SYMPTOM

No character can be printed using either the parallel printer or serial printer.

POSSIBLE CAUSES	SOLUTIONS
• Bad connection between the printer and the computer.	• Check the connectors and cable connecting the printer and the computer.
• Setting of the Parallel / Serial switch on the top cabinet is wrong.	• The switch should be set to the PARALLEL position if parallel printer is used as port 1. The switch should be set to the SERIAL position if serial printer is used.

Circuits to be checked:

a) The DIN socket (J3) and the parallel printer port (J9).

b) The ACIA (U11) and the interfacing IC (U3 and U4).

c) The ACIA clock generator circuit.

d) Pin 60 of gate array 1 (U6).

SYMPTOM

Missing or wrong characters being printed when using either parallel or serial printer.

POSSIBLE CAUSES	SOLUTIONS
• Bad connection between the printer and the computer.	• Check the connector and cable.
• Wrong port configuration.	• Enter the control panel by pressing CTRL-RESET-P and select the correct configuration parameters.
• The user has selected a wrong printer interface through the application software.	• Try other interface options from the list of options provided by the application software.
• Internal hardware failure.	• Check the serial interface IC's and ACIA if serial port is used. Check the parallel interface IC's if parallel port is used.

Circuits to be checked:

a) The DIN socket (J3) and the parallel printer port (J9).

b) The ACIA (U11) and the interfacing IC (U3 and U4).

c) The ACIA clock generator circuit.

d) Pin 60 of gate array 1 (U6).

SYMPTOM

Communication through serial port 2 cannot work properly.

POSSIBLE CAUSES	SOLUTIONS
• Setting of MIDI / MODEM switch on the back panel is wrong in Laser 128 EX/2.	• Set to the MODEM position.
• Bad connection between the serial communication device and the computer.	• Check the connector and the cable.
• Wrong port configuration of serial port 2.	• Enter the control panel by pressing the CTRL-RESET-P and select the correct configuration parameters.
• Internal hardware failure.	• Check the serial interface IC's and ACIA.

Circuits to be checked:

a) The DIN connector (J4).

b) The ACIA (U12) and interface IC (U3 and U4).

c) The ACIA clock generator circuit.

d) Pin 60 of gate array 1 (U6).

SYMPTOM

MIDI port of Laser 128 EX/2 cannot work properly.

POSSIBLE CAUSES	SOLUTIONS
• Bad connection between the MIDI device and the computer.	• Check the connector and the cable.
• Setting of MIDI/ MODEM switch on the back panel is wrong.	• Set to the MIDI position.

SYMPTOM

Peripheral card plugged to the side expansion slot or the Laser expansion box cannot work.

POSSIBLE CAUSES	SOLUTIONS
• Bad contact of card and the computer.	• Clean the gold-plated fingers of the peripheral PCB.
• The setting of the external slot enable switches is wrong.	• The slots enable switches inside the ROM door should be set to enable external slots position.
• The separate power supply of the expansion box is not working.	• Check the transformer AC connection. Check the DC connection from the transformer to the expansion box. • Check the power supply regulator circuit of the expansion box.
• The peripheral card uses the DMA signal while the Laser 128 EX is running at high speed.	• Set the computer speed to the normal 1 MHz.

Circuits to be checked:

a) The 50 way expansion connector (J1).

b) Pin 61, pin 63, pin 72, pin 77 and pin 100 of gate array 1 (U6).

c) The buffer IC (U5 and U13).

SYMPTOM

The built-in expansion RAM cannot be recognised by the computer.

POSSIBLE CAUSES	SOLUTIONS
• The RAM chips are not inserted in the correct bank sequence.	• Insert the RAM chips 8 pieces / bank according to the user manual.
• The RAM is inserted in a wrong direction.	• The RAM should be inserted in the right direction. If it cannot work, new RAM chips have to be replaced the damaged RAM chips.
• Bad inter connection of the RAM board and the main board.	• Make sure the contacts are good and the RAM board should be firmly mounted.
• The setting of external slot 5 switch inside the ROM door is wrong.	• The switch should be set to disable external slot 5 position.

Circuits to be checked:

a) Pin 31 to 44, pin 74 and pin 75 of gate array 3 (U38).

b) U45 to U47 of the expansion RAM board.

SYMPTOM

Game port not functioning properly.

POSSIBLE CAUSES	SOLUTIONS
• Bad connection.	• Check the connector and cable.
• Internal hardware failure.	• Check the game port interface IC's.

Circuits to be checked :

a) Pin 45 to 53 of gate array 3 (U38).

b) Game port connector J10 and the interface IC (U29 to U32 and U34).

D.2 Gate array pin assignments

D.2.1 Gate array 1: Memory Management Unit (HG61H20B48F)

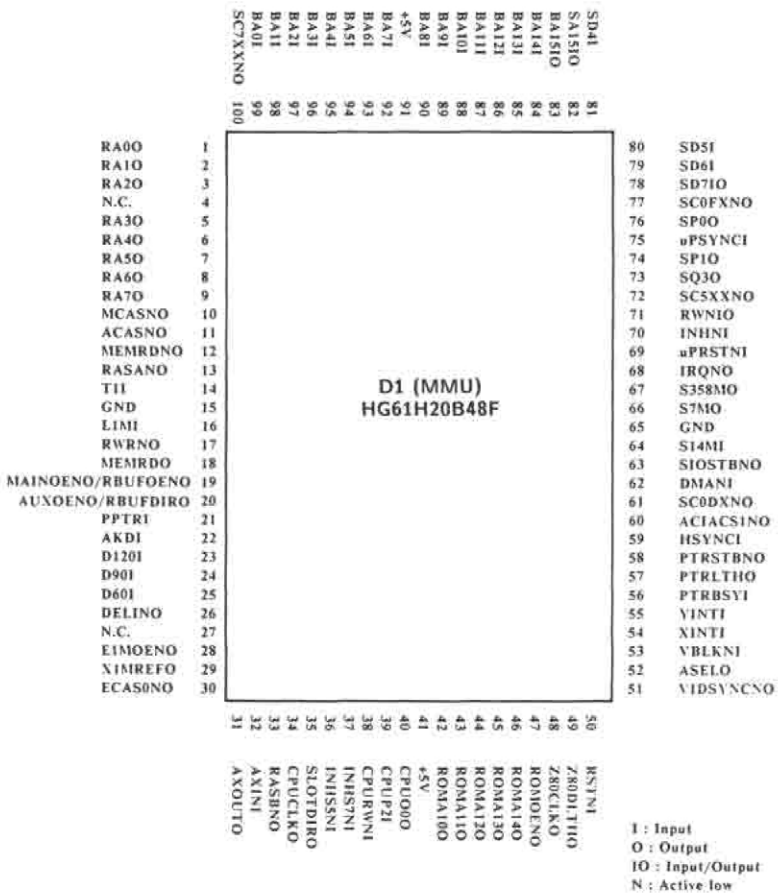


Fig. **D-1** Pin assignment of gate array 1 (MMU)

Pin	Name	Direction	Description
1	RA0	Out	System RAM address bus bit 0
2	RA1	Out	System RAM address bus bit 1
3	RA2	Out	System RAM address bus bit 2
4	N.C.	-	-
5	RA3	Out	System RAM address bus bit 3
6	RA4	Out	System RAM address bus bit 4
7	RA5	Out	System RAM address bus bit 5
8	RA5	Out	System RAM address bus bit 6
9	RA7	Out	System RAM address bus bit 7
10	MCAS-	Out	Main bank system RAM column address strobe

Pin	Name	Direction	Description
11	ACAS-	Out	Auxiliary bank system RAM column address strobe
12	MEMRD-	Out	System RAM output buffer control; low when reading from RAM
13	RASA-	Out	System RAM row address strobe
14	T1	In	Gate array test pin; should be low for normal operation
15	GND	-	System ground
16	L1M	In	Operating mode strapping pin: high for "LASER 128"; low for "LASER 128 EX"
17	RWR-	Out	RAM write-enable signal; low when writing to RAM

Pin	Name	Direction	Description
18	MEMRD	Out	Expansion RAM data bus transceiver direction control; complement of MEMRD- (pin 12)
19	MAINOE-	Out	Main bank system RAM data bus transceiver output buffer control in "LASER 128" (L1M high); low when reading from main RAM
	RBUFOE-	Out	System data bus transceiver output buffer control in "LASER 128 EX" (L1M low)
20	AUXOE-	Out	Auxiliary bank system RAM data bus transceiver output buffer control in "LASER 128" (L1M high); low when reading from auxiliary RAM

Pin	Name	Direction	Description
	RBUFDIR	Out	System data bus transceiver direction control in "LASER 128 EX" (L1M low)
21	PPTR	In	Printer-type selection pin: high for parallel printer; low for serial printer
22	AKD	In	Any-key-down signal generated by 8048 keyboard encoder; high if a key is pressed
23	D120	In	Delay line output (fourth tap)
24	D90	In	Delay line output (third tap)
25	D60	In	Delay line output (second tap)
26	DELIN	Out	Input to delay line
27	N.C.	-	-

Pin	Name	Direction	Description
28	E1MOE-	Out	Expansion RAM data bus transceiver output buffer control; low when accessing expansion RAM
29	X1MREF	Out	Expansion RAM refresh control signal; high during expansion RAM refresh cycle.
30	ECAS0-	Out	Expansion RAM column address strobe
31	AXOUT	Out	Dynamic RAM row/column address select output
32	AXIN	In	Dynamic RAM row/column address select input
33	RASB-	Out	Expansion RAM row address strobe
34	CPUCLK	Out	Clock input to 65C02 CPU
35	SLOTDIR	Out	Unused in current system

Pin	Name	Direction	Description
36	INHS5-	In	Internal/external port 5 select: high for internal port 5 (expansion RAM);
37	INHS7-	In	Internal/external port 7 select: high for internal port 7 (3.5" disk drive interface); low for external port 7
38	CPURW-	In	Read/write line of 65C02 CPU; high in a CPU read cycle; low in a CPU write cycle
39	CPUP2	In	Clock output of 65C02 CPU; delayed version of CPUCLK (pin 34)
40	CPUP0	Out	Gated internally with DMA-(pin 62) to generate CPUCLK (pin 34)

Pin	Name	Direction	Description
41	VDD	-	+5V power supply for the gate array
42	ROMA10	Out	Program ROM address bit 10
43	ROMA11	Out	Program ROM address bit 11
44	ROMA12	Out	Program ROM address bit 12
45	ROMA13	Out	Program ROM address bit 13
46	ROMA14	Out	Program ROM address bit 14
47	ROMOE-	Out	Program ROM output buffer control; low to enable the output buffer
48	Z80CLK	Out	Unused in current system
49	Z80DLTH	Out	Unused in current system
50	RST-	In	System reset signal; low to reset the gate array

Pin	Name	Direction	Description
51	VIDSYNC-	Out	1 MHz (13H:1L) synchronisation signal for VDG gate array
52	ASEL	Out	High when the auxiliary bank system RAM is being accessed
53	VBLK-	In	Vertical blanking signal from the VDG gate array; low during the vertical blanking interval
54	XINT	In	X-direction interrupt signal from the mouse; square waves appear at this pin whenever the mouse moves horizontally
55	YINT	In	Y-direction interrupt signal from the mouse; square waves appear at this pin whenever the mouse moves vertically

Pin	Name	Direction	Description
56	PTRBSY	In	Active-high busy signal from the parallel printer
57	PTRLTH	Out	Parallel printer data latch clock output; printer data latched at rising-edge of this signal
58	PTRSTB-	Out	Parallel printer data strobe output; latched printer data is strobed into the parallel printer at the rising-edge of this signal
59	HSYNC	In	Horizontal display synchronisation signal generated by VDG gate array
60	ACIACS1-	Out	Chip-select output for 6551 ACIAs; low when accessing the two ACIAs

Pin	Name	Direction	Description
61	SC0DX-	Out	Low when accessing addresses from $C0D0 to $C0DF while external port 5 is selected
62	DMA-	In	Low if external interface card is performing direct memory access
63	SIOSTB-	Out	Low when accessing addresses from $C800 to $CFFF while either external port 5 or external port 7 is selected
64	S14M	In	Master clock from which all other timing signals are derived; 14.31818 MHz for NTSC versions; 14.18758 MHz for PAL versions
65	GND	-	System ground
66	S7M	Out	General purpose 7 MHz clock

Pin	Name	Direction	Description
67	S358M	Out	General purpose 3.58 MHz clock
68	IRQ-	Out	Active-low CPU interrupt request output
69	UPRST-	In	CPU reset signal; low during power-up or when CTRL-RESET is pressed
70	INH-	In	Driven low by external interface card to inhibit the on-board system RAM and program ROM (except the I/O firmwares)
71	RW-	Out	When DMA- (pin 62) is high, it is in output mode and is equivalent to CPURW- (pin 38);
		In	when DMA- is low, it is in input mode and is driven by external interface card

Pin	Name	Direction	Description
72	SC5XX-	Out	Low when accessing addresses from $C500 to $C5FF while external port 5 is selected
73	SQ3	Out	General purpose 2 MHz (4H:3L) clock
74	SP1	Out	General purpose 1 MHz clock; complement of SP0 (pin 76)
75	UPSYNC	In	65C02 SYNC signal; high in an opcode-fetch cycle
76	SP0	Out	General purpose 1 MHz clock
77	SC0FX-	Out	Low when accessing addresses from $C0F0 to $C0FF while external port 7 is selected
78	SD7	In/Out	System data bus bit 7
79	SD6	In	System data bus bit 6
80	SD5	In	System data bus bit 5

Pin	Name	Direction	Description
81	SD4	In	System data bus bit 4
82	SA15	In	When DMA- (pin 62) is low, it is in input mode and is driven by the external interface card;
		Out	When DMA- is high, it is in output mode and is a latched version of BA15 (pin 83)
83	BA15	In	System address bus bit 15; When DMA- (pin 62) is high, it is in input mode;
		Out	When DMA- is low, it is in output mode and is equivalent to SA15
84	BA14	In	System address bus bit 14
85	BA13	In	System address bus bit 13

Pin	Name	Direction	Description
86	BA12	In	System address bus bit 12
87	BA11	In	System address bus bit 11
88	BA10	In	System address bus bit 10
89	BA9	In	System address bus bit 9
90	BA8	In	System address bus bit 8
91	VDD	-	+5V power supply for gate array
92	BA7	In	System address bus bit 7
93	BA6	In	System address bus bit 6
94	BA5	In	System address bus bit 5
95	BA4	In	System address bus bit 4

Pin	Name	Direction	Description
96	BA3	In	System address bus bit 3
97	BA2	In	System address bus bit 2
98	BA1	In	System address bus bit 1
99	BA0	In	System address bus bit 0
100	SC7XX-	Out	Low when accessing addresses from $C700 to $C7FF while external port 7 is selected

D.2.2 Gate array 2: Video Display Generator (HG61H20B44F)

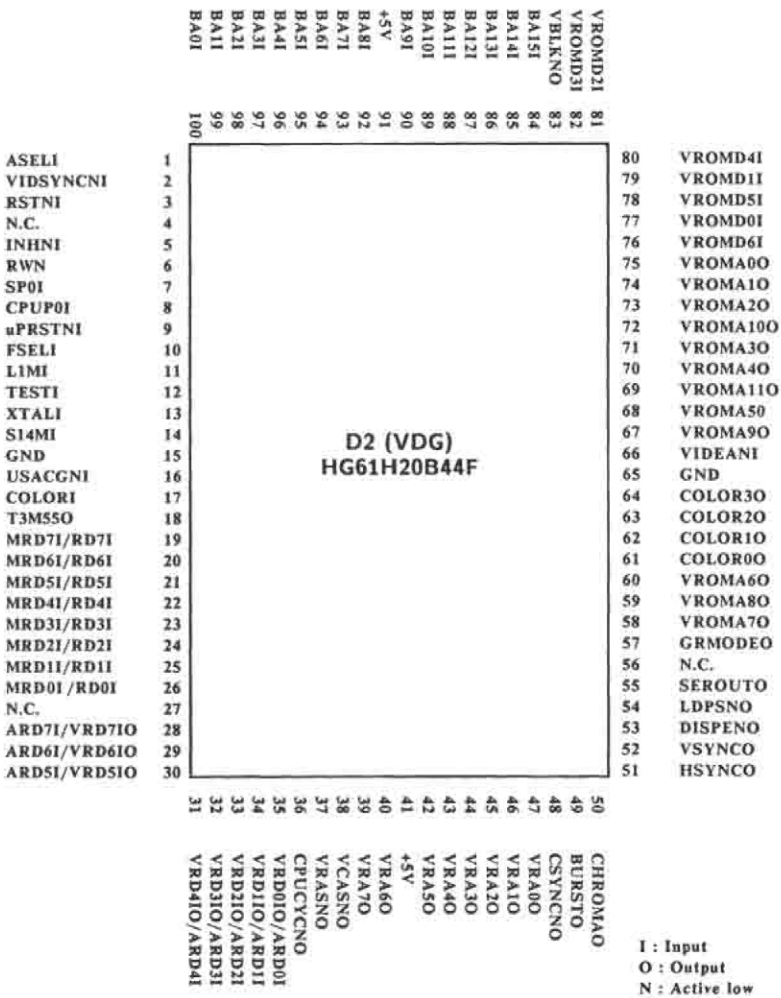


Fig. D-2 Pin assignment of gate array 2 (VDG)

Pin	Name	Direction	Description
1	ASEL	In	High when accessing auxiliary bank system RAM
2	VIDSYNC-	In	1 MHz (13H:1L) synchronisation signal generated by MMU gate array
3	RST-	In	Active-low system reset signal
4	N.C.	-	-
5	INH-	In	Driven low by external interface card to inhibit the video RAM
6	RW-	In	System read/write line; low in a write cycle
7	SP0	In	1 MHz clock
8	CPUP0	In	Non-gated CPU clock generated by MMU gate array
9	UPRST-	In	Active-low CPU reset signal

Pin	Name	Direction	Description
10	FSEL	In	TV system strapping pin: high for NTSC system (60 Hz); low for PAL system (50 Hz)
11	L1M	In	Operating mode strapping pin: high for "LASER 128"; low for "LASER 128 EX"
12	TEST	In	Gate array test pin; should be low in normal operation
13	XTAL	In	Color-subcarrier reference clock; 14.31818 MHz for NTSC versions; 17.73447 MHz for PAL versions
14	S14M	In	Master clock; 14.31818 MHz for NTSC versions; 14.18758 MHz for PAL versions
15	GND	-	System ground

Pin	Name	Direction	Description
16	USACG-	In	Character set strapping pin; low for USA character set; high for foreign character set
17	COLOR	In	Color/monochrome mode selection pin; high for color graphics display
18	T3M55	Out	Not used in current system
19	MRD7	In	Main bank system RAM data bus bit 7 in "LASER 128";
	RD7	In	CPU data bus bit 7 in "LASER 128 EX"
20	MRD6	In	Main bank system RAM data bus bit 6 in "LASER 128";
	RD6	In	CPU data bus bit 6 in "LASER 128 EX"

Pin	Name	Direction	Description
21	MRD5	In	Main bank system RAM data bus bit 5 in "LASER 128";
	RD5	In	CPU data bus bit 5 in "LASER 128 EX";
22	MRD4	In	Main bank system RAM data bus bit 4 in "LASER 128 EX"
	RD4	In	CPU data bus bit 4 in "LASER 128 EX"
23	MRD3	In	Main bank system RAM data bus bit 3 in "LASER 128";
	RD3	In	CPU data bus bit 3 in "LASER 128 EX"
24	MRD2	In	Main bank system RAM dat bus bit 2 in "LASER 128";
	RD2	In	CPU data bus bit 3 in "LASER 128 EX"
25	MRD1	In	Main bank system RAM data bus bit 1 in "LASER 128";

Pin	Name	Direction	Description
	RD1	In	CPU data bus bit 1 in "LASER 128 EX"
26	MRD0	In	Main bank system RAM data bus bit 0 in "LASER 128";
	RD0	In	CPU data bus bit 0 in "LASER 128 EX"
27	N.C.	-	-
28	ARD7	In	Auxiliary bank system RAM data bus bit 7 in "LASER 128";
	VRD7	In/Out	Video RAM data bus bit 7 in "LASER 128 EX"
29	ARD6	In	Auxiliary bank system RAM data bus bit 6 in "LASER 128";
	VRD6	In/Out	Video RAM data bus bit 6 in "LASER 128 EX"
30	ARD5	In	Auxiliary bank system RAM data bus bit 5 in "LASER 128";

Pin	Name	Direction	Description
	VRD5	In/Out	Video RAM data bus bit 5 in "LASER 128 EX"
31	ARD4	In	Auxiliary bank system RAM data bus bit 4 in "LASER 128";
	VRD4	In/Out	Video RAM data bus bit 4 in "LASER 128 EX"
32	ARD3	In	Auxiliary bank system RAM data bus bit 3 in "LASER 128";
	VRD3	In/Out	Video RAM data bus bit 3 in "LASER 128 EX"
33	ARD2	In	Auxiliary bank system RAM data bus bit 2 in "LASER 128";
	VRD2	In/Out	Video RAM data bus bit 2 in "LASER 128 EX"
34	ARD1	In	Auxiliary bank system RAM data bus bit 1 in "LASER 128";

Pin	Name	Direction	Description
	VRD1	In/Out	Video RAM data bus bit 1 in "LASER 128 EX"
35	ARD0	In	Auxiliary bank system RAM data bus bit 0 in "LASER 128";
	VRD0	In/Out	Video RAM data bus bit 0 in "LASER 128 EX"
36	CPUCYC-	Out	Video RAM write-enable signal in "LASER 128 EX"; low during a RAM write cycle; not used in "LASER 128"
37	VRAS-	Out	Video RAM row address strobe in "LASER 128 EX"; not used in "LASER 128"
38	VCAS-	Out	Video RAM column address strobe in "LASER 128 EX"; not used in "LASER 128"

Pin	Name	Direction	Description
39	VRA7	Out	Video RAM address bus bit 7 in "LASER 128 EX"; System RAM address bus bit 7 in "LASER 128"
40	VRA6	Out	Video RAM address bus bit 6 in "LASER 128 EX"; System RAM address bus bit 6 in "LASER 128"
41	VDD	-	+5V power supply for the gate array
42	VRA5	Out	Video RAM address bus bit 5 in "LASER 128 EX"; System RAM address bus bit 5 in "LASER 128"
43	VRA4	Out	Video RAM address bus bit 4 in "LASER 128 EX"; System RAM address bus bit 4 in "LASER 128"

Pin	Name	Direction	Description
44	VRA3	Out	Video RAM address bus bit 3 in "LASER 128 EX"; System RAM address bus bit 3 in "LASER 128"
45	VRA2	Out	Video RAM address bus bit 2 in "LASER 128 EX"; System RAM address bus bit 2 in "LASER 128"
46	VRA1	Out	Video RAM address bus bit 1 in "LASER 128 EX"; System RAM address bus bit 1 in "LASER 128"
47	VRA0	Out	Video RAM address bus bit 0 in "LASER 128 EX"; System RAM address bus bit 0 in "LASER 128"

Pin	Name	Direction	Description
48	CSYNC-	Out	Composite synchronisation signal for generating composite video
49	BURST	Out	Used for generating the color burst in the composite video
50	CHROMA	Out	Used for generating the chrominence signal in a composite video signal
51	HSYNC	Out	Horizontal synchronisation signal for TV or monitor
52	VSYNC	Out	Vertical synchronisation signal for TV or monitor
53	DISPEN-	Out	Active-high composite blanking signal for the LCD display
54	LDPS-	Out	Timing synchronisation signal for the LCD display
55	SEROUT	Out	Serial data output for the LCD display

Pin	Name	Direction	Description
56	N.C.	-	-
57	GRMODE	Out	Video ROM address bit 12; high when displaying graphics; low when displaying text
58	VROMA7	Out	Video ROM address bit 7
59	VROMA8	Out	Video ROM address bit 8
60	VROMA6	Out	Video ROM address bit 6
61	COLOR0	Out	Digital color code bit 0
62	COLOR1	Out	Digital color code bit 1
63	COLOR2	Out	Digital color code bit 2
64	COLOR3	Out	Digital color code bit 3
65	GND	-	System ground

Pin	Name	Direction	Description
66	VIDEA-	In	Must be low for normal operation
67	VROMA9	Out	Video ROM address bit 9
68	VROMA5	Out	Video ROM address bit 5
69	VROMA11	Out	Video ROM address bit 11
70	VROMA4	Out	Video ROM address bit 4
71	VROMA3	Out	Video ROM address bit 3
72	VROMA10	Out	Video ROM address bit 10
73	VROMA2	Out	Video ROM address bit 2
74	VROMA1	Out	Video ROM address bit 1
75	VROMA0	Out	Video ROM address bit 0
76	VROMD6	In	Video ROM data bit 6

Pin	Name	Direction	Description
77	VROMD0	In	Video ROM data bit 0
78	VROMD5	In	Video ROM data bit 5
79	VROMD1	In	Video ROM data bit 1
80	VROMD4	In	Video ROM data bit 4
81	VROMD2	In	Video ROM data bit 2
82	VROMD3	In	Video ROM data bit 3
83	VBLK-	Out	Active-low vertical blanking signal; 60 Hz for NTSC versions; 50 Hz for PAL versions
84	BA15	In	System address bus bit 15
85	BA14	In	System address bus bit 14
86	BA13	In	System address bus bit 13
87	BA12	In	System address bus bit 12
88	BA11	In	System address bus bit 11

Pin	Name	Direction	Description
89	BA10	In	System address bus bit 10
90	BA9	In	System address bus bit 9
91	VDD	-	+5V power supply for the gate array
92	BA8	In	System address bus bit 8
93	BA7	In	System address bus bit 7
94	BA6	In	System address bus bit 6
95	BA5	In	System address bus bit 5
96	BA4	In	System address bus bit 4
97	BA3	In	System address bus bit 3
98	BA2	In	System address bus bit 2

Pin	Name	Direction	Description
99	BA1	In	System address bus bit 1
100	BA0	In	System address bus bit 0

D.3 Gate array 3: Universal Disk Controller (HG61H20B56F)

Fig. D-3 Pin assignment of gate array 3 (UDC)

Pin	Name	Direction	Description
1	RAM-	Out	Chip-enable for disk buffer 6116; low when accessing RAM
2	RAMA10	Out	Disk buffer address bit 10
3	PHA	Out	Stepper motor phase 0 control
4	N.C.	-	-
5	PHB	Out	Stepper motor phase 1 control
6	PHC	Out	Stepper motor phase 2 control
7	PHD	Out	Stepper motor phase 3 control
8	RDATA	In	Disk drive read data
9	WDATA	Out	Disk drive serial write data
10	WGATE-	Out	Write-gate signal for disk drive; low when writing to the drive

Pin	Name	Direction	Description
11	SIDSEL	Out	Side-select signal for double-side drives
12	ENAB1-	Out	Low when drive 1 is enabled
13	ENAB2-	Out	Low when drive 2 is enabled
14	INTSEP-	In	Internal/external data separator strapping pin; must be low in current system
15	GND	-	System ground
16	INTC7-	In	Operating mode strapping pin; must be low in current system
17	WP	In	Write-protect signal from disk drive; high if disk is write-protected
18	PWM	Out	Spindle motor speed control for 3.5" disk drives; constant frequency but varying duty cycle

Pin	Name	Direction	Description
19	KBDSTB-	In	Active-low strobe generated by the keyboard encoder whenever a key is pressed
20	PPTR	In	Printer-type strapping pin: low for serial printer; high for parallel printer
21	SPKR	Out	Speaker control signal
22	KBD6	In	Bit 6 of key code generated by the keyboard encoder 8048
23	KBD5	In	Bit 5 of key code
24	KBD4	In	Bit 4 of key code
25	KBD3	In	Bit 3 of key code
26	KBD2	In	Bit 2 of key code
27	N.C.	-	-
28	KBD1	In	Bit 1 of key code
29	KBD0	In	Bit 0 of key code

Pin	Name	Direction	Description
30	ERA8	Out	Expansion RAM address bit 8
31	ERA7	Out	Expansion RAM address bit 7
32	ERA6	Out	Expansion RAM address bit 6
33	ERA5	Out	Expansion RAM address bit 5
34	ERA4	Out	Expansion RAM address bit 4
35	ERA3	Out	Expansion RAM address bit 3
36	ERA2	Out	Expansion RAM address bit 2
37	ERA1	Out	Expansion RAM address bit 1
38	ERA0	Out	Expansion RAM address bit 0
39	ECAS1-	Out	Column address strobe for first 256K-byte expansion RAM

Pin	Name	Direction	Description
40	ECAS2-	Out	Column address strobe for second 256 K-byte expansion RAM
41	VDD	-	+5V power supply for the gate array
42	ECAS3-	Out	Column address strobe for third 256 K-byte expansion RAM
43	ECAS4-	Out	Column address strobe for fourth 256 K-byte expansion RAM
44	RASB-	In	Expansion RAM row address strobe
45	GI0	In	"40/80 column" switch status; high if switch is in "40" position
46	GI1	In	Game switch 0 status; high if switch is closed
47	GI2	In	Game switch 1 status; high if switch is closed

Pin	Name	Direction	Description
48	GI3	In	Mouse signature; low if mouse is connected to the game port
49	GI4	In	556 timer output for analog input 0; a positive pulse will be generated after accessing any locations from $C070 to $C07F
50	GI5	In	556 timer output for analog input 1; a positive pulse will be generated after accessing any locations from $C070 to $C07F
51	GI6	In	Mouse XDIR output; square waves will be produced whenever the mouse moves horizontally
52	GI7	In	Mouse YDIR output; square waves will be produced whenever the mouse moves vertically

Pin	Name	Direction	Description
53	C07X-	Out	556 timer trigger pulse; goes low whenever the locations from $C070 to $C07F are accessed
54	L1MRA4	Out	System RAM address bus bit 4 in "LASER 128"; not used in "LASER 128 EX"
55	DR35	Out	High when selecting 3.5" drive; low when 5.25" drive is selected
56	DR525EN1-	Out	Low when internal 5.25" drive is enabled
57	SEPD-	In	Separated disk data; must be low in current system
58	SEPCLK	In	Separated data window; must be low in current system
59	ACIACK	Out	1.8 MHz clock for 6551 ACIAs in NTSC versions; not used in PAL versions

Pin	Name	Direction	Description
60	SP0	In	1 MHz general purpose clock
61	SQ3	In	2 MHz general purpose clock
62	RA4	In	Connect to RA4 of MMU and VRA4 of VDG in "LASER 128"; not used in "LASER 128 EX"
63	S14M	In	Master clock; 14.31818 MHz for NTSC versions; 14.18758 MHz for PAL versions
64	S7M	In	General purpose 7 MHz clock
65	GND	-	System ground
66	INHS7-	In	Internal/External port 7 strapping pin: High for internal port 7 (3.5" drive interface); low for external port 7

Pin	Name	Direction	Description
67	CPUP0	In	Non-gated CPU clock
68	READY-	Out	65C02 CPU ready control; low to extend current read cycle
69	UPRST-	In	Active-low CPU reset signal
70	RST-	In	Active-low power-up reset signal
71	AX	In	Dynamic RAM row/column address select input
72	RW-	In	System read/write signal; low in a write cycle
73	INHS5-	In	Internal/external port 5 strapping pin: High for internal port 5 (expansion RAM interface); low for external port 5

Pin	Name	Direction	Description
74	ECAS0-	In	Decoded internally to generate the column address strobe signals for the expansion RAM
75	X1MREF	In	High to initiate an expansion RAM refresh cycle
76	SD7	In/Out	System data bus bit 7
77	SD6	In/Out	System data bus bit 6
78	SD5	In/Out	System data bus bit 5
79	SD4	In/Out	System data bus bit 4
80	SD3	In/Out	System data bus bit 3
81	SD2	In/Out	System data bus bit 2
82	SD1	In/Out	System data bus bit 1
83	SD0	In/Out	System data bus bit 0
84	BA15	In	System address bus bit 15
85	BA14	In	System address bus bit 14

Pin	Name	Direction	Description
86	BA13	In	System address bus bit 13
87	BA12	In	System address bus bit 12
88	BA11	In	System address bus bit 11
89	BA10	In	System address bus bit 10
90	BA9	In	System address bus bit 9
91	VDD	-	+5V power supply for the gate array
92	BA8	In	System address bus bit 8
93	BA7	In	System address bus bit 7
94	BA6	In	System address bus bit 6
95	BA5	In	System address bus bit 5
96	BA4	In	System address bus bit 4

Pin	Name	Direction	Description
97	BA3	In	System address bus bit 3
98	BA2	In	System address bus bit 2
99	BA1	In	System address bus bit 1
100	BA0	In	System address bus bit 0

D.3 Parts lists

D.3.1 Laser 128

DESTINATION		PART NUMBER	DESCRIPTION
SW1,SW2		42-0032-00-00	SWITCH 2P2T 103436280
	V	42-0032-00-02	SLIDE SWITCH MODEL NO. SSSS322NA2-
SW4		42-0125-01-00	ROCKER SWITCH 3P2T ART NO. 110E
J11		40-0415-20-04	FLAT CABLE CONNECTOR 2 X 10 PINS
	V	40-0415-20-05	FLAT CABLE CONNECTOR 20 WAYS
	A	40-0200-01-00	FLAT CABLE CONNECTOR 2 X 10 PINS
	A	40-0200-01-01	FLAT CABLE CONNECTOR 2 X 10 PINS
	A	40-0200-02-00	FLAT CABLE CONNECTOR 2 X 10 PINS
J1		40-0632-05-01	EDGE CONN. 50 PINS RAT-050-210
	R	40-0632-05-00	EDGE CONN. 50 PINS RAT-050-210
J2		40-0457-07-00	DIN PLUG 7 PINS DJ-014-PT
J3,J4		40-0459-05-00	DIN PLUG 7 PINS DJ-021-5PS
J5		40-0669-19-01	D-SUB CONN. 19 PINS FEMALE DMR-19S
	R	40-0456-19-02	D-TYPE CONN. 19 PINS FEMALE DMR-19
	R	40-0456-19-00	DB-19 FEMALE SOCKET 19 PINS RIGHT
J7,J9		40-0669-15-01	D-SUB CONN. 15 PINS FEMALE DMR-15S
	R	40-0456-15-02	D-TYPE CONN. 15 PINS FEMALE DMR-15
	R	40-0456-15-00	DB-15 FEMALE SOCKET 15 PINS
J10		40-0669-09-01	D-SUB CONN. 9 PINS FEMALE DMR-9S-Y
	R	40-0456-09-02	FEMALE SOCKET 9 PINS DMR-9S-W
	R	40-0456-09-00	DB-9 FEMALE SOCKET 9 PINS
J13		40-0255-30-06	DOUBLE ROW HEADER 2 X 15 PINS
	A	40-0611-03-00	DUAL ROW HEADER 2 X 15 PINS DRW-30
J14		40-0132-00-00	EARPHONE JACK HSJ 0922-01-010
	A	40-0684-00-00	PHONE JACK 5 PINS SCJ-0356A-5P
J6		40-0133-01-00	PIN JACK JPJ 0544-01-310
	A	40-0133-02-01	RCA JACK LP-0844
	A	40-0133-00-00	PIN JACK J00545-01-210
U20		27-0460-00-00	G65SC02P-1 (1MHZ)
	A	27-0685-00-00	VL65NC02-02PC (2MHZ)
U6		27-0690-00-00	GATE ARRAY HG61H20B48F D1
U25		27-0691-00-00	GATE ARRAY HG61H20B44E D2
U38		27-0692-00-00	GATE ARRAY HG61H20B56F D3.2
U15		27-0473-02-04	EPROM (32K X 8) (250NS) TMS27C256
	A	27-0473-00-00	EPROM (32K X 8) (250NS)
	A	27-0473-00-01	EPROM (32K X 8) (250NS) HN27256G-2
	A	27-0473-00-02	EPROM (32K X 8) (250NS)
	A	27-0473-00-03	EPROM (32K X 8) (250NS)
	A	27-0473-00-04	EPROM (32K X 8) (250NS)
U35,U36,U39		27-0597-01-01	DRAM (64K X 4) TMM41464P-12
U40	V	27-0597-01-02	DRAM (64K X 4) (120NS) MT4067-12
	V	27-0597-01-00	DRAM (64K X 4) (120NS) HM50464P-12
	V	27-0597-01-03	DRAM (64K X 4) (120NS) UPD41464-12
	A	27-0597-00-00	DRAM (64K X 4) (150NS) HM50464P-15
	A	27-0597-00-01	DRAM (64K X 4) (150NS) TMM41464P-1
	A	27-0597-00-02	DRAM (64K X 4) (150NS) MSM41464RS-
	A	27-0597-00-03	DRAM (64K X 4) (150NS) UPO41464-15
	A	27-0597-00-04	DRAM (64K X 4) (150NS) TMS4464-15
U18		27-0057-00-03	I.C. 74LS05
	V	27-0057-00-02	I.C. 74LS05
	V	27-0057-00-01	I.C. 74LS05
	V	27-0057-00-04	I.C. 74LS05
	V	27-0057-00-05	I.C. 74LS05

DESTINATION		PART NUMBER	DESCRIPTION
U26		27-0038-02-00	I.C. 74LS04N
	V	27-0038-02-05	I.C. 74LS04N
	V	27-0038-02-01	I.C. 74LS04N
	V	27-0038-02-02	I.C. 74LS04
	V	27-0038-02-03	I.C. 74LS04
U21		27-0440-00-00	I.C. RCA74HCT374
	V	27-0440-00-01	I.C. SN74HCT374N
	V	27-0440-00-02	I.C. MC74HCT374
	V	27-0440-00-03	I.C. 74HCT374P
	V	27-0440-00-04	I.C. 74HCT374
U34		27-0253-00-07	DUAL TIMING CIRCUITS UA556
	V	27-0253-00-06	DUAL TIMER NE556N
	V	27-0253-00-01	I.C. NE556
	V	27-0253-00-02	I.C. NE556
U22		27-0145-00-00	I.C. 74LS145
	V	27-0145-00-01	I.C. 74LS145
	V	27-0145-00-02	I.C. 74LS145N
U21		27-0703-00-00	SINGLE CHIP MICROCOMPUTER 8048AH
	A	27-0687-00-00	LSI UPD80C48C
U14		27-0706-00-00	MASK ROM 64K VT37-0706-0 (200NS)
	A	27-0143-02-00	EPROM 2764 (250NS)
	A	27-0143-02-01	EPROM (8K X 8) M5L2764K (250NS)
	A	27-0143-02-02	EPROM (8K X 8) M2764-25 (250NS)
U37		27-0133-06-01	CMOS SRAM (2K X 8) HY6116P-12 (120
	V	27-0133-06-02	CMOS SRAM (2K X 8) CXK5816PN-12 (1
	A	27-0133-04-02	SRAM (2K X 8) LC3516P-20 (200NS)
	A	27-0133-05-04	RAM (2K X 8) HY6116-15 (150NS)
	A	27-0133-05-05	CMOS SRAM (2K X 8) KM6816-15 (150N
	A	27-0133-05-07	CMOS SRAM (2K X 8) V62C16P15
	A	27-0133-05-08	CMOS SRAM (2K X 8) UM6116-1 (150NS
	A	27-0133-07-00	SRAM HM6116LP-2 (120NS)
	A	27-0133-07-01	CMOS SRAM (2K X 8) HM6116L-5 (120N
U41,U44		27-0436-01-00	I.C. KS74HCTLS74A
	A	27-0436-00-00	I.C. RCA74HCT74
	A	27-0436-00-02	I.C. PC74HCT74P
U5,U13,U19,		27-0441-00-00	I.C. RCA74HCT244
U23,U27	V	27-0441-00-03	I.C. PC74HCT244P
	V	27-0441-00-05	I.C. M74HCT244
U11,U12		27-0458-00-00	I.C. 6551 (1MHZ)
	A	27-0656-00-00	I.C. S6551 (1.8432MHZ)
	A	27-0656-01-00	I.C. S6551AP (1.8432MHZ)
U14		27-0363-00-05	QUAD LINE DRIVER XR1488
	V	27-0363-00-02	I.C. LINER DFIVER MC1488
	V	27-0363-00-00	I.C. MC1488
	V	27-0363-00-01	I.C. MC1488
	V	27-0363-00-01	I.C. MC1488
	V	27-0363-00-03	I.C. SN75188
	V	27-0363-00-04	I.C. GD75188

DESTINATION		PART NUMBER	DESCRIPTION
U3		27-0364-00-05	QUAD LINE RECEIVER XR-1489A
	V	27-0364-00-00	I.C. MC1489
	V	27-0364-00-01	I.C. MC1489
	V	27-0364-00-02	I.C. MC1489A
	V	27-0364-00-03	I.C. SN75189
	V	27-0364-00-04	I.C. GD75189A
U2		27-0614-00-01	I.C. TL494
	V	27-0614-00-00	I.C. TL494CN
U28,U42,U43		27-0614-00-00	I.C. KS74HCTLS245
	A	27-0438-02-00	I.C. KS74AHCT245
	A	27-0438-00-00	I.C. RCA74HCT245
	A	27-0438-00-01	I.C. 74HCT245
	A	27-0438-00-03	I.C. 74HCT245
	A	27-0438-00-04	I.C. 74HCT245
L127		25-1065-01-00	CHOKE COIL 68UH +/-10%
	A	25-1065-00-00	COIL 68UH +/-5%
L94-L96		25-1065-00-00	CHOKE 32UH 6010-6H-3TS
L1,L7-15,L17,		25-1109-00-00	3 1/2T FERRITE BEAD CHOKE HOR. MOU
L20,L22-L31, L33-L43, L47-L51,L55, L56,L60,L64, L80,L86-92, L100-L102, L104,L105, L126,L128	V	25-1109-00-10	3 1/2T FERRITE BEAD CHOKE HOR. MOU
L57-L59,L63		25-1108-00-01	2 1/2T FERRITE BEAD CHOKE HOR. MOU
	R	25-1103-00-00	2 1/2T FERRITE BEAD CHOKE VERT. MO
L19,L44-L46,		25-4004-00-01	FERRITE BEAD CHOKE RH 3.5 X 5 X 1.
L53,L53,L61, L62,L83-L85	V	25-4004-00-00	FERRITE BEAD CHOKE RH 3.5 X 5 X 1.
L2-L6,L16,		25-1104-01-00	3 1/2T FERRITE BEAD CHOKE VERT. MO
L18,L21,L32, L106-L114	A	25-1104-00-00	3 1/2T FERRITE BEAD CHOKE VERT. MO
L81,L93		25-1022-01-00	CHOKE COIL 1MH +/-5%
	A	25-1022-00-00	CHOKE COIL 1MH +/-10%
L82		25-1020-00-00	CHOKE COIL 3.3H
	V	25-1020-00-02	CHOKE COIL 3.3H +/-10%
	V	25-1020-00-03	CHOKE COIL 3.3H +/-10%
L129,L130		25-1189-00-00	FERRITE BEAD CHOKE 1 1/2T
CM3		25-1072-00-00	FILTER CHOKE COIL TK-4716
	V	25-1072-00-01	FILTER CHOKE COIL TMP-1228
CM2		25-1173-00-01	COMMON MODE CHOKE 3.7MH +/-20% @1K
	V	25-1173-00-00	COMMON MODE CHOKE 3.7MH +/-20% @1K
	A	25-1201-00-00	COMMON MODE CHOKE 25T RC0207/88
	A	25-1175-00-00	COMMON MODE CHOKE 16T VERT. MOUNTI
CM1		52-2284-20-00	COMMON MODE CHOKE C014/C015 20T VE
T2		52-2284-14-00	X'FORMER EE-19 5V/-12V
	A	52-2284-08-00	TOROIDAL X'FORMER 5V/-12V
	A	52-2284-11-00	X'FORMER EI-19 5V/-12V
XTAL1		25-3014-01-01	CRYSTAL 14.31818 MHZ
	V	25-3015-01-00	CRYSTAL 14.31818 MHZ
XTAL3		25-3-45-01-00	CRYSTAL 3.6864 MHZ

DESTINATION		PART NUMBER	DESCRIPTION
VR2		24-2502-00-00	TRIMMER RESISTOR 5K OHM RVF8P01A50
VR4		24-2202-00-00	TRIMMER RESISTOR 2K OHM RVF8P01-20
VR3		24-0012-01-00	VOLUME D50K OHM 16SN01
	V	24-0012-01-01	VOLUME 50K OHM
	A	24-0053-00-00	VOLUME 20K OHM, TYPE B
	A	24-0104-00-00	ROTARY POTENTIOMETER 20K OHM 16SN-
R89		23-0011-10-02	RESISTOR 10 OHM +/-5% 1/4W
	V	23-0011-10-00	RESISTOR 10 OHM +/-5% 1/4W
R14,R73,R74,		23-0012-10-02	RESISTOR 100 OHM +/-5% 1/4W
R113,R14	V	23-0012-10-00	RESISTOR 100 OHM +/-5% 1/4W
R43,R45,R49,		23-0013-10-02	RESISTOR 1K OHM +/-5% 1/4W
R54,R64,R81,	V	23-0013-10-00	RESISTOR 1K OHM +/-5% 1/4W
R84,R85,R88,			
R90,R91,R93,			
R101,R108,			
R133			
R25,R30,R71,		23-0014-10-02	RESISTOR 10K OHM +/-5% 1/4W
R72	V	23-0014-10-00	RESISTOR 10K OHM +/-5% 1/4W
	A	23-0014-13-00	RESISTOR 10K OHM +/-2% 1/4W
R16		23-0015-10-02	RESISTOR 100K OHM +/-5% 1/4W
	V	23-0015-10-00	RESISTOR 100K OHM +/-5% 1/4W
R19		23-0017-10-02	RESISTOR 10M OHM +/-5% 1/4W
	V	23-0017-10-00	RESISTOR 10M OHM +/-5% 1/4W
R21		23-0122-10-02	RESISTOR 1.2K OHM +/-5% 1/4W
	V	23-0122-10-00	RESISTOR 1.2K OHM +/-5% 1/4W
R31,R32,R134		23-0123-10-02	RESISTOR 12K OHM +/-5% 1/4W
R15		23-0151-33-02	RESISTOR 150 OHM +/-5% 1W
	V	23-0151-33-02	RESISTOR 150 OHM +/-5% 1W
R18,R44		23-0152-10-02	RESISTOR 1.5K OHM +/-5% 1/4W
	V	23-0152-10-00	RESISTOR 1.5K OHM +/-5% 1/4W
R27,R31,R32		23-0153-10-02	RESISTOR 15K OHM +/-5% 1/4W
	V	23-0153-10-00	RESISTOR 15K OHM +/-5% 1/4W
R75,R82		23-0181-10-02	RESISTOR 180 OHM +/-5% 1/4W
	V	23-0181-10-00	RESISTOR 180 OHM +/-5% 1/4W
R26		23-0183-10-02	RESISTOR 18K OHM +/-5% 1/4W
	V	23-0183-10-00	RESISTOR 18K OHM +/-5% 1/4W
R99		23-0220-10-02	RESISTOR 22 OHM +/-5% 1/4W
	V	23-0220-10-00	RESISTOR 2 OHM +/-5% 1/4W
R76,R81		23-0221-10-02	RESISTOR 220 OHM +/-5% 1/4W
	V	23-0221-10-00	RESISTOR 220 OHM +/-5% 1/4W
R55,R130		23-0222-10-02	RESISTOR 2.2K OHM +/-5% 1/4W
	V	23-0222-10-00	RESISTOR 2.2K OHM +/-5% 1/4W
R83,R86		23-0223-10-02	RESISTOR 22K OHM +/-5% 1/4W
	V	23-0223-10-00	RESISTOR 22K OHM +/-5% 1/4W
R77,R80		23-0271-10-02	RESISTOR 270 OHM +/-5% 1/4W
	V	23-0271-10-00	RESISTOR 270 OHM +/-5% 1/4W
R103		23-0272-10-02	RESISTOR 2.7K OHM +/-5% 1/4W
	V	23-0272-10-00	RESISTOR 2.7K OHM +/-5% 1/4W
R135,R136,		23-0330-10-02	RESISTOR 33 OHM +/-5% 1/4W
R138,R140	V	23-0330-10-00	RESISTOR 33 OHM +/-5% 1/4W
R56,R57,R98		23-0331-10-02	RESISTOR 330 OHM +/-5% 1/4W
	V	23-0331-10-00	RESISTOR 330 OHM +/-5% 1/4W

DESTINATION		PART NUMBER	DESCRIPTION
R33,R34,R35, R37,R38,R40, R41,R42,R46, R47,R48,R50, R51,R53,R58, R59,R60,R62, R66,R67,R69, R70,R94,R95, R111,R112, R190		23-0332-10-02	RESISTOR 3.3K OHM +/-5% 1/4W
	V	23-0332-10-02	RESISTOR 3.3K OHM +/-5% 1/4W
R20,R28,R39		23-0333-10-02	RESISTOR 33K OHM +/-5% 1/4W
	V	23-0333-10-02	RESISTOR 33K OHM +/-5% 1/4W
R61,R63,R65, R68		23-0391-10-00	RESISTOR 390 OHM +/-5% 1/4W
	V	23-0391-10-00	RESISTOR 390 OHM +/-5% 1/4W
R23,R92		23-0392-10-02	RESISTOR 3.9K OHM +/-5% 1/4W
	V	23-0392-10-02	RESISTOR 3.9K OHM +/-5% 1/4W
R17		23-0399-10-02	RESISTOR 3.9 OHM +/-5% 1/4W
	V	23-0399-10-00	RESISTOR 2.9 OHM +/-5% 1/4W
R78,R79		23-0431-10-02	RESISTOR 430 OHM +/-5% 1/4W
	V	23-0431-10-00	RESISTOR 430 OHM +/-5% 1/4W
R24,R52		23-0471-10-02	RESISTOR 470 OHM +/-5% 1/4W
	V	23-0471-10-02	RESISTOR 470 OHM +/-55 1/4W
R104,R116		23-0472-10-02	RESISTOR 4.7K OHM +/-5% 1/4W
	V	23-0472-10-00	RESISTOR 4.7K OHM +/-5% 1/4W
R107		23-0473-10-02	RESISTOR 47K OHM +/-5% 1/4W
	V	23-0473-10-00	RESISTOR 47K OHM +/-5% 1/4W
R87		23-0561-10-02	RESISTOR 560 OHM +/-5% 1/4W
	V	23-0561-10-00	RESISTOR 560 OHM +/-5% 1/4W
R97		23-0681-10-02	RESISTOR 680 OHM +/-5% 1/4W
	V	23-0681-10-00	RESISTOR 680 OHM +/-5% 1/4W
R102,R105		23-0682-10-02	RESISTOR 6.8K OHM +/-5% 1/4W
	V	23-0682-10-00	RESISTOR 6.9K OHM +/-5% 1/4W
C44,C51,C59		22-1100-21-71	ELEC CAP 10UF 16V +/-20%
	V	22-1100-21-03	ELEC CAP 10UF 16V +/-20%
	A	22-1100-11-03	ELEC CAP 10UF 10V +/-20%
	A	22-1100-13-00	ELEC CAP 10UF 10V +50-10%
	A	22-1100-13-11	ELEC CAP 10UF 10V +50-10%
	V	22-1100-21-55	ELEC CAP 10UF 16V +/-20%
	V	22-1100-21-56	ELEC CAP 10UF 16V +/-20%
	V	22-1100-21-58	ELEC CAP 10UF 16V +/-20%
C15		22-1100-31-00	ELEC CAP 10UF 25V +/-20%
	A	22-1100-33-00	ELEC CAP 10UF 25V +50-10%
	A	22-1100-33-11	ELEC CAP 10UF 25V +50-10%
C17,C41		22-1470-61-70	ELEC CAP 47UF 50V +/-20%
	V	22-1470-61-03	ELEC CAP 47UF 50V +/-20%
	A	22-1101-61-03	ELEC CAP 100UF 50V +/-20%
C8		22-1221-31-70	ELEC CAP 220UF 25V +/-20%
	V	22-1221-31-03	ELEC CAP 220UF 25V +/-20%
	V	22-1221-31-55	ELEC CAP 220UF 25V +/-20%
	V	22-1221-31-56	ELEC CAP 220UF 25V +/-20%
	V	22-1221-31-58	ELEC CAP 220UF 25V +/-20%
	A	22-1221-33-15	ELEC CAP 220UF 25V +50-10%
	A	22-1221-33-41	ELEC CAP 220UF 25V +50-10%

DESTINATION		PART NUMBER	DESCRIPTION
C30,C31,C32		22-1221-11-70	ELEC CAP 220UF 10V +/-20%
	V	22-1221-11-03	ELEC CAP 220UF 10V +/-20%
	V	22-1221-11-56	ELEC CAP 220UF 10V +/-20%
	V	22-1221-11-58	ELEC CAP 220UF 10V +/-20%
	A	22-1221-13-11	ELEC CAP 220UF 10V +50-10%
C1		22-1221-21-03	ELEC CAP 220UF 16V +/-20%
C60		22-1229-61-71	ELEC CAP 2.2UF 50V +/-20%
	V	22-1229-61-03	ELEC CAP 2.2UF 50V +/-20%
	V	22-1229-61-04	ELEC CAP 2.2UF 50V +/-20%
	V	22-1229-61-55	ELEC CAP 2.2UF 50V +/-20%
	V	22-1229-61-56	ELEC CAP 2.2UF 50V +/-20%
	V	22-1229-61-58	ELEC CAP 2.2UF 50V +/-20%
C26,C27,C28,		22-1229-21-00	ELEC CAP 100UF 16V +/-20%
C193	V	22-1101-21-55	ELEC CAP 100UF 16V +/-20%
	V	22-1101-21-56	ELEC CAP 100UF 16V +/-20%
	V	22-1101-21-58	ELEC CAP 100UF 16V +/-20%
	V	22-1101-21-58	ELEC CAP 100UF 16V +/-20%
	A	22-1101-23-15	ELEC CAP 100UF 16V +50-10%
	A	22-1101-23-41	ELEC CAP 100UF 16V +50-10%
C25,C45,C57		22-1220-21-71	ELEC CAP 22UF 16V +/-20%
	V	22-1220-21-03	ELEC CAP 22UF 16V +/-20%
	V	22-1220-21-55	ELEC CAP 22UF 16V +/-20%
	V	22-1220-21-56	ELEC CAP 22UF 16V +/-20%
	V	22-1220-21-58	ELEC CAP 22UF 16V +/-20%
C23		22-1109-61-71	ELEC CAP 1UF 50V +/-20%
	V	22-1109-61-03	ELEC CAP 1UF 50V +/-20%
	V	22-1109-61-55	ELEC CAP 1UF 50V +/-20%
	V	22-1109-61-56	ELEC CAP 1UF 50V +/-20%
	V	22-1109-61-58	ELEC CAP 1UF 50V +/-20%
C7		22-1220-41-03	ELEC CAP 22UF 35V +/-20%
	V	22-1220-41-55	ELEC CAP 22UF 35V +/-20%
	V	22-1220-41-56	ELEC CAP 22UF 35V +/-20%
	V	22-1220-41-58	ELEC CAP 22UF 35V +/-20%
C46		22-1470-11-71	ELEC CAP 47UF 10V +/-20%
	V	22-1470-11-03	ELEC CAP 47UF 10V +/-20%
	V	22-1470-11-55	ELEC CAP 47UF 10V +/-20%
	V	22-1470-11-56	ELEC CAP 47UF 10V +/-20%
	V	22-1470-11-58	ELEC CAP 47UF 10V +/-20%
	A	22-1470-13-15	ELEC CAP 47UF 10V +50-10%
	A	22-1470-13-41	ELEC CAP 47UF 10V +50-10%
C50,C53,C62,		22-3104-28-74	CER CAP 0.1UF 50V +80-20%
C63,C65,C289,	V	22-3104-28-00	CER CAP 0.1UF 50V +80-20%
C290	A	22-3120-18-00	CER CAP 0.1UF 25V +80-20%
	A	22-3104-18-15	CER CAP 0.1UF 25V +80-20%
C24,C66-C75,		22-3120-28-69	CER CAP 1000PF 50V +80-20%
C79,C83-C88,	V	22-3102-28-15	CER CAP 1000PF 50V +80-20%
C91,C92,C96,	V	22-3102-28-00	CER CAP 1000PF 50V +80-20%
C100-C112,	V	22-3102-28-65	CER CAP 1000PF 50V +80-20%
C114,C116,	A	22-3102-26-00	CER CAP 1000PF 50V +/-10%
C138-C144,	A	22-3102-27-00	CER CAP 1000PF 50V +/-20%
C160-C191,			
C200,C272,			
C301			

DESTINATION		PART NUMBER	DESCRIPTION
C49,		22-3121-26-74	CER CAP 120PF 50V +/-10%
C252-C261,	V	22-3121-26-00	CER CAP 120PF 50V +/-10%
C267-C273			
C35-C38		22-3331-26-74	CER CAP 330PF 50V +/-10%
	V	22-3331-26-00	CER CAP 330PF 50V +/-10%
C274		22-3390-26-00	CER CAP 39PF 50V +/-10%
C47,C48,		22-3100-26-69	CER CAP 10PF 50V +/-10%
C277	V	22-3100-26-00	CER CAP 10PF 50V +/-10%
	A	22-3100-21-00	CER CAP 10PF 50V +/-5%
	A	22-3100-25-00	CER CAP 10PF 50V +/-5%
	A	22-3100-25-15	CER CAP 10PF 50V +/-5%
C268		22-3181-26-74	CER CAP 180PF 50V +/-10%
	V	22-3181-26-00	CER CAP 180PF 50V +/-10%
C2-C6,C29,		22-3403-26-74	CER CAP 0.04UF 50V +/-10%
C33,C34,C43,	A	22-3473-18-15	CER CAP 0.047UF 25V +80/-20%
C52,C57,C58,	A	22-3403-28-00	CER CAP 0.04UF 50V +80-20%
C61,C64,	A	22-3403-28-02	CER CAP 0.04UF 50V +80-20%
C76-C78,C113,	A	22-3403-28-19	CER CAP 0.04UF 50V +80-20%
C118-C127,			
C130-C132,			
C145-C156,			
C194,C201,			
C202,C262-C264,			
C266,C278			
C158,C159		22-3203-28-00	CER CAP 0.02UF 50V +80-20%
	A	22-3203-26-00	CER CAP 0.02UF 50V +/-10%
C90,C93-C95,		22-3101-26-69	CER CAP 100PF 50V +/-10%
C97-C99,	V	22-3101-26-00	CER CAP 100PF 50V +/-10%
C169,C245,	A	22-3101-25-00	CER CAP 100PF 50V +/-5%
C250,C251,	A	22-3101-25-09	CER CAP 100PF 50V +/-5%
C244			
C287,C300		22-3680-26-00	CER CAP 68PF +/-10%
C54,C55		22-3223-25-72	MONO CAP 0.022UF 50V +/-5%
	V	22-3223-25-36	MONO CAP 0.022UF 50V +/-5%
	V	22-3223-25-13	MONO CAP 0.022UF 50V +/-5%
	V	22-3223-25-37	MONO CAP 0.022UF 50V +/-5%
C13		22-6103-26-01	MYLER CAP 0.01UF 50V +/-10%
	A	22-6103-46-40	MYLER CAP 0.01UF 100V +/-10%
C195,C198		22-6473-46-47	MYLER CAP 0.01UF 100V +/-10%
	A	22-6473-26-01	MYLER CAP 0.047UF 50V +/-10%
VC1		22-7002-01-00	TRIMMER CAP 4PF-20PF +80-0% CTC-6U
	A	22-7002-00-00	TRIMMER CAP 20PF
D4,D7-D14,		21-0001-00-00	DIODE 1N4148
D19,D20	V	21-0001-00-02	DIODE 1N4148
D2		21-0158-00-01	SCHOTTKY DIODE 1N5821 30V 3A
	V	21-0158-00-00	SCHOTTKY DIODE 1N5821 30V 3A
	A	21-0167-00-00	SCHOTTKY BARRIER RECT. MBR340P 40V
D16,D17		21-0032-00-00	DIODE 2-1K60
Z1		21-0042-00-00	ZENER DIODE 5.1V @ IZ=5MA 0.4W BZX
Z2		21-0039-02-00	ZENER DIODE 13V IZ=10MA MTZ13B
	A	21-0039-01-00	ZENER DIODE 13V 500MW +/-20% BZX79

DESTINATION		PART NUMBER	DESCRIPTION
Z7		21-0019-04-00	ZENER DIODE 6.8V IZ=20MA MTZ6.8C
	A	21-0019-00-00	ZENER DIODE 6.8V 500MA @ IZ=5MA
	A	21-0019-00-01	ZENER DIODE 6.8V 500MW @ IZ=5MA
	A	21-0019-00-01	ZENER DIODE 6.8V 500MW @ IZ=5MA
Z3-Z6		21-0006-05-00	ZENER DIODE MT25.6#B 20MA
	A	21-0006-03-00	ZENER DIODE 5.6V +/-2% 500MW @ IZ=
Q18		20-0011-02-01	TRANSISTOR 9018H
Q7,Q8		20-0014-01-01	TRANSISTOR 9012G
	A	20-0014-00-01	TRANSISTOR 9012H
Q11,Q12		20-0019-00-01	TRANSISTOR 9014C
	V	20-0019-00-00	TRANSISTOR 9014C
	V	20-0019-00-05	TRANSISTOR H9014C
	A	20-0025-03-00	TRANSISTOR 1402D
Q17		20-0025-01-02	TRANSISTOR 1402B
	V	20-0025-01-01	TRANSISTOR ST 1402B
Q5,Q6,Q14, Q15		20-0025-03-00	TRANSISTOR 1402D
Q9,Q10		20-0003-02-00	TRANSISTOR 8050C
	A	20-0028-02-00	TRANSISTOR NA31XJ
	A	20-0028-03-00	TRANSISTOR NA31XI/J
	A	20-0028-04-00	TRANSISTOR NA31XI/J/H
Q4,Q13		20-0033-01-00	TRANSISTOR ST1602D
Q3		20-0112-00-00	TRANSISTOR BD438
	A	20-0042-00-00	TRANSISTOR BD434
Q1,Q2		20-0105-00-00	PNP POWER TRANSISITOR TIP42
	A	20-0105-00-01	POWER TRANSISTOR TIP42
	A	20-0105-01-00	PNP TRANSISTOR TIP42C
	A	20-0105-02-00	PNP POWER TRANSISTOR TIP42A
	A	20-0105-03-00	PNP POWER TRANSISTOR TIP42B
U29-U32		20-0044-00-02	OPTO-COUPLER 4N27
	V	20-0044-00-00	OPTO-COUPLER 4N27

D.3.2 Laser 128EX

DESTINATION		PART NUMBER	DESCRIPTION
SW1,2		42-0032-00-00	SWITCH
	V	42-0032-00-02	SLIDE SWITCH
SW4		42-0125-01-00	ROCKER SWITCH
J11		40-0415-20-04	FLAT CABLE CONNECTOR 2 X 10 PIN
	V	40-0415-20-05	FLAT CABLE CONN.
	A	40-0200-01-00	FLAT CABLE CONN.
	A	40-0200-01-01	FLAT CABLE CONN.
	A	40-0200-02-00	FLAT CABLE CONN.
J1		40-0632-50-01	EDGE CONN 50 PINS
	R	40-0632-50-00	EDGE CONN 50 PINS
J2		40-0457-07-00	DIN PLUG 7PINS
J3,4		40-0459-05-00	DIN SOCKET 5 PINS
J5		40-0669-19-01	D-SUB CONN. 19 PINS
	R	40-0456-19-02	D-TYPE CONN
	R	40-0456-19-00	DB19 FEMALE SOCKET
J7,9		40-0669-15-01	D-SUB CONN. 15 PINS
	R	40-0456-15-02	D-TYPE CONN.
	R	40-0456-15-00	DB-15 FEMALE SOCKET
J10		40-0699-09-01	D-SUB CONN 9 PINS
	R	40-0456-09-02	FEMALE SOCKET
	R	40-0456-09-00	DB-9 FEMALE SOCKET
J6		40-0133-01-00	PIN JACK
	A	40-0133-02-01	RCA JACK
	A	40-0133-00-00	PIN JACK
J14		40-0132-00-00	EARPHONE JACK
	A	40-0684-00-00	PHONE JACK
J13		40-0255-30-06	DOUBLE ROW HEADER
	A	40-0611-30-00	DUAL ROW HEADER
U20		27-0695-00-00	CPU G65SC02P-4 (4MHZ)
U6		27-0690-00-00	GATE ARRAY 1 (HG61H20B48F)
U25		27-0691-00-00	GATE ARRAY 2 (HG61H20B44F)
U38		27-0692-00-00	GATE ARRAY 3 (HG61H20B56F)
U15		27-0473-02-04	EPROM (32K X 8) (250NS) TMS27C2
	V	27-0473-02-02	EPROM (32K X 8) (250NS) NMC27C2
	A	27-0473-00-00	EPROM 27256 256K (250NS)
	A	27-0473-00-01	EPROM 27256 256K (250NS) (HN272
	A	27-0473-00-02	EPROM 27256 256K (250NS)
	A	27-0473-00-03	EPROM 27256 256K (250NS)
	A	27-0473-00-04	EPROM 27256 256K (250NS)
U7,8,9,10		27-0597-01-01	DRAM (64K X 4) TMM41464P-12
	V	27-0597-01-02	DRAM (64K X 4) MT4067-12
	V	27-0597-01-00	DRAM HM50464P-12
	V	27-0597-01-03	DRAM (64K X 4) UPD41464-12
U35,36		27-0597-01-01	DRAM (64K X 4) TMM41464P-12
	V	27-0597-01-00	DRAM HM50464P-12 (120NS)
	V	27-0597-01-03	DRAM UPD41464-12 (120NS)
	A	27-0597-00-00	DRAM HM50464P-15 (150NS)
	A	27-0597-00-01	DRAM TMM41464P-15 (150NS)
	A	27-0597-00-02	DRAM MSM41464RS (150NS)
	A	27-0597-00-03	DRAM UP041464-15 (150NS)

DESTINATION		PART NUMBER	DESCRIPTION
U5		27-0160-00-00	74LS244
	V	27-0160-00-02	HD74LS244
	V	27-0160-00-03	74LS244N
	V	27-0160-00-05	74LS244
	V	27-0160-00-08	GD74LS244
U13,27		27-0441-00-00	RCA74HCT244
	V	27-0441-00-03	PC74HCT244P
	V	27-0441-00-05	M74HCT244
U11,12		27-0458-00-00	6551 (1MHZ)
	A	27-0656-00-00	S6551 (ACIA - 1.8432MHZ)
	A	27-0656-01-00	S6551AP (1.8432MHZ)
U4		27-0363-00-05	RS-232 QUAD LINE DRIVER XR1448
	V	27-0363-00-02	LINEAR DRIVER (SIGNETICS) MC144
	V	27-0363-00-00	MC1448
	V	27-0363-00-01	MC1448
	V	27-0363-00-03	SN75188
	V	27-0363-00-04	GD75188
U3		27-0364-00-05	RS-232 QUAD LINE RECEIVER XR-14
	V	27-0364-00-00	MC1489
	V	27-0364-00-01	MC1489
	V	27-0364-00-02	LINE RECEIVER (SIGNETICS) MC148
	V	27-0364-00-03	SN75189
	V	27-0364-00-04	GD75189A
U2		27-0614-00-01	TL494
	V	27-0614-00-00	TL494CN
U17		27-0438-01-00	KS74HCTLS245
	A	27-0438-02-00	KS74AHCT245
	A	27-0438-00-00	RCA74HCT245
	A	27-0438-00-01	74HCT245
	A	27-0438-00-03	74HCT245
	A	27-0438-00-04	74HCT245
L127		25-1065-01-00	CHOKE COIL 68UH
	A	25-1065-00-00	COIL 68UH
L94-96		25-1082-00-00	CHOKE 32UH
CM3		25-1072-00-00	FILTER CHOKE COIL
	V	25-1072-00-01	FILTER CHOKE
L2-6,32,104-114		25-1104-01-00	3 1/2 T FERRITE BEAD CHOKE VERT
	A	25-1104-00-00	3 1/2 T FERRITE BEAD CHOKE VERT
	A	25-1104-00-01	3 1/2 T FERRITE BEAD CHOKE VERT
L7-18,20,22		25-1109-00-00	3 1/2 T FERRITE BEAD CHOKE HORI
24-31,33-43	V	25-1109-00-01	3 1/2 T FERRITE BEAD CHOKE HORI
47-51,55,56			
60,63-80			
86-92,100-102			
110,126,128			
L19,44-46,53,54		25-4004-00-01	FRRRITE BEAD RH3.5X5X1.2MM
61,62,83-85	V	25-4004-00-00	FERRITE BEAD RH3.5X5X1.2MM
L129-131		25-1189-00-00	FERRITE BEAD CHOKE
	R	25-4044-00-00	FERRITE MULTI HOLE CORE
L63,57-59		25-1108-00-01	2 1/2 T FERRITE BEAD CHOKE HORI
	R	25-1103-00-00	2 1/2 T FERRITE BEAD CHOKE VERT
L81,93,103		25-1022-01-00	CHOKE COIL
	A	25-1022-00-00	CHOKE COIL

DESTINATION		PART NUMBER	DESCRIPTION
U18		27-0057-00-03	74LS05
	V	27-0057-00-02	74LS05
	V	27-0057-00-01	74LS05
	V	27-0057-00-04	74LS05
	V	27-0057-00-05	74LS05
U26		27-0038-02-00	74LS04N
	V	27-0038-02-01	74LS04N
	V	27-0038-02-02	74LS04
	V	27-0038-02-03	74LS04
	V	27-0038-02-05	74LS04N
U24		27-0183-00-00	74LS373
	V	27-0183-00-01	74LS373
	V	27-0183-00-02	74LS373N
	V	27-0183-00-06	74LS373
	V	27-0183-00-07	74LS373
	V	27-0183-00-09	GD74LS373
	A	27-0452-00-00	RCA74HCT373
	A	27-0452-00-01	74HCT373
	A	27-0452-00-02	74HCT373
	A	27-0452-00-03	74HCT373
	A	27-0452-00-04	74HCT373
	A	27-0452-01-00	KS74HCTLS373
U34		27-0253-00-07	DUAL TIMING CIRCUITS UA556
	V	27-0253-00-06	DUAL TIMER NE556N
	V	27-0253-00-01	NE556 (T.I.)
	V	27-0253-00-02	NE556 (THOMSON)
U22		27-0145-00-00	74LS145
	V	27-0145-00-01	74LS145
	V	27-0145-00-02	74LS145N
U21		27-0703-00-00	8048AH K.B. ENCODER
	A	27-0687-00-00	UPD80C48C (6MHZ, 1K X 8 BITS RO
U14		27-0706-00-00	64K MASK ROM VT27-0706-0 (200NS
	A	27-0143-02-00	EPROM 2764 (250NS)
	A	27-0143-02-01	EPROM M5L2764K (250NS)
	A	27-0143-02-02	EPROM M2764-25 (250NS)
U41		27-0470-01-00	KS74HCTLS175
	A	27-0470-00-01	74HCT175
	A	27-0470-00-00	RCA74HCT175
U39,80		27-0436-01-00	KS74HCTLS74A
	A	27-0436-00-00	RCA74HCT74
	A	27-0436-00-02	PC74HCT74P
U40		27-0443-01-00	KS74HCTLS32
	A	27-0443-00-00	RCA74HCT32
	A	27-0443-00-01	74HCT32
U37		27-0133-06-01	CMOS SRAM (2K X 8) HY6116P-12 (
	V	27-0133-06-02	CMOS SRAM (2K X 8) CXK5816PN-12
	A	27-0133-04-03	SRAM (2K X 8) LC3516P-20 (200NS
	A	27-0133-05-04	RAM (2K x 8) HY6116-15 (150NS)
	A	27-0133-05-05	CMOS SRAM (2K X 8) KM6816-15 (1
	A	27-0133-05-07	CMOS SRAM (2K X 8) V62C16P15
	A	27-0133-05-08	CMOS SRAM (2k X 8) UM6116-1 (15
	A	27-0133-07-00	STATIC RAM (2K X 8) HM6116LP-2
	A	27-0133-07-01	CMOS SRAM (2K X 8) HM6116L-5 (1

DESTINATION		PART NUMBER	DESCRIPTION
L82,97		25-1020-00-00	CHOKE COIL
	V	25-1020-00-02	CHOKE COIL
	V	25-1020-00-03	CHOKE COIL
CM2		25-1173-00-01	COMMON MODE CHOKE 3.7MH +/-20% @1KHZ 29T, 0.8MM DIA
	V	25-1173-00-00	COMMON MODE CHOKE 3.7MH +/-20% @1KHZ 29T, 0.8MM DIA
	A	25-1201-00-00	COMMON MODE CHOKE RC0207/88 DIA 0.8MM WIRE 25 TURNS
	A	25-1175-00-00	COMMON MODE CHOKE 16T, 0.8MM, 3 VERTICAL MOUNT
T2		52-2284-14-00	EE-19 X'FORMER, 5V/-12V, C014/1 2E1, 24:55T, I SAT > 4 AMP @PRI
	A	52-2284-08-00	TOROIDAL X'FORMER (5V/-12V) INSTEAD OF 52-2284-07-00
	A	52-2284-11-00	EI-19 X'FORMER 5V/-12V, C014/15 2E6, 24:55T, I SAT > 4 AMP @ PR
CM1		52-2284-20-00	COMMON MODE CHOKE (C014/C015) 2 P/N:25-4033-00-00, VERTICAL
XTAL1		25-3015-01-01	CRYSTAL 14.31818 MHZ
	A	25-3015-01-00	CRYSTAL 14.31818 MHZ
XTAL3		25-3045-01-00	CRYSTAL 3.6864MHZ
U16		25-6007-00-00	DELAY LINE T46A111
	V	25-6007-00-01	DELAY LINE EP9810-110
	V	25-6007-00-02	DELAY LINE DL2996
VR2		24-2502-00-00	TRIMMER RESISTOR 5K OHM
VR4		24-2202-00-00	TRIMMER RESISTOR 2K OHM
VR3		24-0012-01-00	VOLUME 50K 16MM
	V	24-0012-01-01	VOLUME 50K 16MM
	A	24-0053-00-00	VOLUMN 20K 16MM
	A	24-0104-00-00	ROTARY POTENTIOMETER 20K
R14,45,73,74		23-0012-10-02	RESISTOR 100 OHM TAPE
113,114	V	23-0012-10-00	RESISTOR 100 OHM
R43,54,55,64		23-0013-10-02	RESISTOR IK OHM TAPE
84,85,88,90	V	23-0013-10-00	RESISTOR IK OHM
91,93,101			
108,133			
R25,30,71,82		23-0014-10-02	RESISTOR 10K OHM TAPE
	V	23-0014-10-00	RESISTOR 10K OHM
	A	23-0014-13-00	RESISTOR 10K OHM
R16		23-0015-10-02	RESISTOR 100K OHM TAPE
	V	23-0015-10-00	RESISTOR 100K OHM
R19		23-0017-10-02	RESISTOR 10M OHM TAPE
	V	23-0017-10-00	RESISTOR 10M OHM
R21		23-0122-10-02	RESISTOR 1.2K OHM TAPE
	V	23-0122-10-00	RESISTOR 1.2K OHM
R131,132,134		23-0123-10-02	RESISTOR 12K OHM TAPE
R15		23-0151-33-02	METAL FILM RESISTOR TAPE 150 OH
	V	23-0151-33-00	METAL FILM RESISTOR 150 OHM
R18,27		23-0152-10-02	RESISTOR 1.5K OHM TAPE
	V	23-0152-10-00	RESISTOR 1.5K OHM
R31,32		23-0153-10-02	RESISTOR 15K OHM TAPE
	V	23-0153-10-00	RESISTOR 15K OHM

DESTINATION		PART NUMBER	DESCRIPTION
R75,82		23-0181-10-02	RESISTOR 180 OHM TAPE
	V	23-0181-10-00	RESISTOR 180 OHM
R26		23-0183-10-02	RESISTOR 18K OHM TAPE
	V	23-0183-10-00	RESISTOR 18K OHN
R99		23-0220-10-02	RESISTOR 22 OHM TAPE
	V	23-0220-10-00	RESISTOR 22 OHM
R76,81		23-0221-10-02	RESISTOR 220 OHM TAPE
	V	23-0221-10-00	RESISTOR 220 OHM
R49,130		23-0222-10-02	RESISTOR 2.2K OHM TAPE
	V	23-0222-10-00	RESISTOR 2.2K OHM
R83,86		23-0223-10-02	RESISTOR 22K OHM TAPE
	V	23-0223-10-00	RESISTOR 22K OHM
R77,80		23-0271-10-02	RESISTOR 270 OHM TAPE
	V	23-0217-10-00	RESISTOR 270 OHM
R103		23-0272-10-02	RESISTOR 2.7K OHM TAPE
	V	23-0272-10-00	RESISTOR 2.7K OHM
R135,136,137		23-0330-10-02	RESISTOR 33 OHM TAPE
138,139	V	23-0330-10-00	RESISTOR 33 OHM
R56,57,98		23-0331-10-02	RESISTOR 330 OHM TAPE
	V	23-0331-10-00	RESISTOR 330 OHM
R33,34,35,36		23-0332-10-02	RESISTOR 3.3K OHM TAPE
37,40,41,42	V	23-0332-10-00	RESISTOR 3.3K OHM
46,47,48,50			
51,53,58,59			
60,62,66,67			
70,94,95,109			
R20,28,39		23-0333-10-02	RESISTOR 33K OHM TAPE
	V	23-0333-10-00	RESISTOR 33K OHM
R61,63,65,68		23-0391-10-02	RESISTOR 390 OHM TAPE
	V	23-0391-10-00	RESISTOR 390 OHM
R23,92		23-0392-10-02	RESISTOR 3.9K OHM TAPE
	V	23-0392-10-00	RESISTOR 3.9K OHM
R17		23-0399-10-02	RESISTOR 3.9 OHM TAPE
	V	23-0399-10-00	RESISTOR 3.9 OHM
R78,79		23-0431-10-02	RESISTOR 430 OHM TAPE
	V	23-0431-10-00	RESISTOR 430 OHM
R24,52		23-0471-10-02	RESISTOR 470 OHM TAPE
	V	23-0471-10-00	RESISTOR 470 OHM
R104,111		23-0472-10-02	RESISTOR 4.7K OHM TAPE
	V	23-0472-10-00	RESISTOR 4.7K OHM
R107		23-0473-10-02	RESISTOR 47K OHM TAPE
	V	23-0473-10-00	RESISTOR 47K OHM
R87		23-0561-10-02	RESISTOR 560 OHM TAPE
	V	23-0561-10-00	RESISTOR 560 OHM
R97		23-0681-10-02	RESISTOR 680 OHM TAPE
	V	23-0681-10-00	RESISTOR 680 OHM
R102,105		23-0682-10-02	RESISTOR 6.8K OHM TAPE
	V	23-0682-10-00	RESISTOR 6.8K OHM
R29		23-0019-39-00	METAL FILM RESISTOR 0.1 OHM +/-
	A	23-0019-33-00	METAL FILM RESISTOR 0.1 OHM +/-
	A	23-0019-38-00	METAL FILM RESISTOR 0.1 OHM +/-

DESTINATION		PART NUMBER	DESCRIPTION
C44,51,59		22-1100-21-71	ELEC CAP 10UF +/-20%, 16V
	V	22-1100-21-03	ELEC CAP 10UF +/-20%, 16V U-TYP
	A	22-1100-11-03	ELEC CAP 10UF +/-20%, 10V
	A	22-1100-13-00	ELEC CAP 10UF +50/-10%, 10V
	A	22-1100-13-11	ELEC CAP 10UF +50/-10%, 10V SSM
	A	22-1100-21-55	ELEC CAP 10UF +/-20%, 16V, RB S
	A	22-1100-21-56	ELEC CAP 10UF +/-20%, 16V, GR S
	A	22-1100-21-58	ELEC CAP 10UF +/-20%, 16V, RS S
C15		22-1100-31-00	ELEC CAP 10UF +/-20%, 25V NLA S
	A	22-1100-33-00	ELEC CAP 10UF +50 -10% 25V U-TY
	A	22-1100-33-11	ELEC CAP 10UF +50 -10% 25V
C17		22-1470-61-70	ELEC CAP 47UF +/-20%, 50V
	V	22-1470-61-03	ELEC CAP 47UF +/-20%, 50V
	A	22-1101-61-03	ELEC CAP 100UF +/-20%, 50V
C1		22-1221-25-11	ELEC CAP 220UF +80/-20%, 16V
	A	22-1221-21-03	ELEC CAP 220UF +/-20%, 16V
	A	22-1221-21-55	ELEC CAP 220UF +/-20%, 16V
	A	22-1221-21-56	ELEC CAP 220UF +/-20%, 16V
	A	22-1221-21-58	ELEC CAP 220UF +/-20%, 16V
C8		22-1221-31-70	ELEC CAP 220UF +/-20%, 25V
	V	22-1221-31-03	ELEC CAP 220UF +/-20%, 25V
	V	22-1221-31-55	ELEC CAP 220UF +/-20%, 25V
	V	22-1221-31-56	ELEC CAP 220UF +/-20%, 25V
	V	22-1221-31-58	ELEC CAP 220UF +/-20%, 25V
	A	22-1221-33-15	ELEC CAP 220UF +50-10%, 25V
	A	22-1221-33-41	ELEC CAP 220UF +50-10%, 25V
C30,31,32		22-1221-11-70	ELEC CAP 220UF +/-20%, 10V
	V	22-1221-11-03	ELEC CAP 220UF +/-20%, 10V
	V	22-1221-11-56	ELEC CAP 220UF +/-20%, 10V
	V	22-1221-11-58	ELEC CAP 220UF +/-20%, 10V
	A	22-1221-13-11	ELEC CAP 220UF +50-10%, 10V
C26,27,28,193		22-1101-21-00	ELEC CAP 100UF +/-20%, 16V
	V	22-1101-21-55	ELEC CAP 100UF +/-20%, 16V
	V	22-1101-21-56	ELEC CAP 100UF +/-20%, 16V
	V	22-1101-21-58	ELEC CAP 100UF +/-20%, 16V
	A	22-1101-23-15	ELEC CAP 100UF +50-10%, 16V
	A	22-1101-23-41	ELEC CAP 100UF +50-10%, 16V
C25		22-1220-21-71	ELEC CAP 22UF +/-20%, 16V
	V	22-1220-21-03	ELEC CAP 22UF +/-20%, 16V
	V	22-1220-21-55	ELEC CAP 22UF +/-20%, 16V
	V	22-1221-21-56	ELEC CAP 22UF +/-20%, 16V
	V	22-1220-21-58	ELEC CAP 22UF +/-20%, 16V
C39,45		22-1220-21-71	ELEC CAP 22UF +/-20%, 16V
	V	22-1220-21-03	ELEC CAP 22UF +/-20%, 16V
	V	22-1220-21-55	ELEC CAP 22UF +/-20%, 16V
	V	22-1221-21-56	ELEC CAP 22UF +/-20%, 16V
	V	22-1220-21-58	ELEC CAP 22UF +/-20%, 16V
	V	22-1220-21-11	ELEC CAP 22UF +/-20%, 16V
	A	22-1220-11-04	ELEC CAP 22UF +/-20%, 16V
	A	22-1220-23-00	ELEC CAP 22UF +50-10%, 16V

DESTINATION		PART NUMBER	DESCRIPTION
C57		22-1220-11-04	ELEC CAP 22UF +/-20%, 10V
	V	22-1220-21-03	ELEC CAP 22UF +/-20%, 16V
	V	22-1220-21-55	ELEC CAP 22UF +/-20%, 16V
	V	22-1221-21-56	ELEC CAP 22UF +/-20%, 16V
	V	22-1220-21-58	ELEC CAP 22UF +/-20%, 16V
C60		22-1229-61-71	ELEC CAP 2.2UF +/-20%, 50V
	V	22-1229-61-03	ELEC CAP 2.2UF +/-20%, 50V
	V	22-1229-61-04	ELEC CAP 2.2UF +/-20%, 50V
	V	22-1229-61-55	ELEC CAP 2.2UF +/-20%, 50V
	V	22-1229-61-56	ELEC CAP 2.2UF +/-20%, 50V
	V	22-1229-61-58	ELEC CAP 2.2UF +/-20%, 50V
C7		22-1220-41-03	ELEC CAP 22UF +/-20%, 35V
	V	22-1220-41-55	ELEC CAP 22UF +/-20%, 35V
	V	22-1220-41-56	ELEC CAP 22UF +/-20%, 35V
	V	22-1220-41-58	ELEC CAP 22UF +/-20%, 35V
C23		22-1109-61-71	ELEC CAP 1UF +/-20%, 50V
	V	22-1109-61-03	ELEC CAP 1UF +/-20%, 50V
	V	22-1109-61-55	ELEC CAP 1UF +/-20%, 50V
	V	22-1109-61-56	ELEC CAP 1UF +/-20%, 50V
	V	22-1109-61-58	ELEC CAP 1UF +/-20%, 50V
C41		22-1470-11-71	ELEC CAP 47UF +/-20%, 10V
	V	22-1470-11-03	ELEC CAP 47UF +/-20%, 10V
	V	22-1470-11-55	ELEC CAP 47UF +/-20%, 10V
	V	22-1470-11-56	ELEC CAP 47UF +/-20%, 10V
	V	22-1470-11-58	ELEC CAP 47UF +/-20%, 10V
	A	22-1470-13-15	ELEC CAP 47UF +50-10%, 10V
	A	22-1470-13-41	ELEC CAP 47UF +50-10%, 10V
C62,63,65		22-3104-28-74	CER CAP 0.1UF +80-20%, 50V
264,289,290	V	22-3104-28-00	CER CAP 0.1UF +80-20%, 50V
	A	22-3104-18-00	CER CAP 0.1UF +80-20%, 25V
	A	22-3104-18-15	CER CAP 0.1UF +80-20%, 25V
C24,66-75,79		22-3102-28-69	CER CAP 1000PF +80-20%, 50V
83-88,91,92	V	22-3102-28-15	CER CAP 1000PF +80-20%, 50V
96,100-112,114	V	22-3102-28-00	CER CAP 0.001UF +80-20%, 50V
116,138-144	V	22-3102-28-65	CER CAP 1000PF +80-20%, 50V
160-191,200	A	22-3102-26-00	CER CAP 1000PF +80-20%, 50V
271,272	A	22-3102-27-00	CER CAP 1000PF +/-20%, 50V
C283		22-3181-26-74	CER CAP 180PF +/-10%, 50V
	V	22-3181-26-00	CER CAP 180PF +/-10%, 50V
C35,36,37,38		22-3331-26-74	CER CAP 330PF +/-10%, 50V
	V	22-3331-26-00	CER CAP 330PF -/-10%, 50V
C274		22-3390-26-00	CER CAP 39PF +/-10%, 50V
C47,48,277		22-3100-26-69	CER CAP 10PF +/-10%, 50V
	V	22-3100-26-00	CER CAP 10PF +/-10%, 50V
	A	22-3100-21-00	CER CAP 10PF +/- 0.5PF, 50V
	A	22-3100-25-00	CER CAP 10PF +/- 5%, 50V
	A	22-3100-25-15	CER CAP 10PF +/-5%, 50V
C261,273		22-3121-26-74	CER CAP 120PF +/-10%, 50V
	V	22-3121-26-00	CER CAP 120PF +/-10%, 50V

DESTINATION		PART NUMBER	DESCRIPTION
C46		22-3470-26-69	CER CAP 47PF +/-10%, 50V
	V	22-3470-26-00	CER CAP 47PF +/-10%, 50V
	V	22-3470-26-01	CER CAP 47PF +/-10%, 50V
	A	22-3470-25-00	CER CAP 47PF +/-5%, 50V
C2-6,29,33,34		22-3403-26-74	CER CAP 0.04UF +/-10%, 50V
40,43,50,52	A	22-3473-18-15	CER CAP 0.047UF +80/-20%, 25V
53,56,58,64	A	22-3403-28-00	CER CAP 0.04UF +80-20% 50V
76,77,78,113	A	22-3403-28-02	CER CAP 0.04UF +80/-20%, 50V
118-128	A	22-3403-28-19	CER CAP 0.04UF +80-20%, 50V
130-132,145-149			
151-156,194,199			
201,202,246,263			
266,275,278,284			
C158,159		22-3203-28-00	CER CAP 0.02UF +80-20%, 50V
	A	22-3203-26-00	CER CAP 0.02UF +/-10%, 50V
C90,93-95,97		22-3101-26-69	CER CAP 100PF +/-10%, 50V
98,99,196,244	V	22-3101-26-00	CER CAP 100PF +/-10%, 50V
245,250,251	A	22-3101-25-00	CER CAP 100PF +/-5%, 50V
282	A	22-3101-25-09	CER CAP 100PF +/-5%, 50V
C49		22-3104-28-70	MONO CAP 0.1UF +80-20%, 50V
	V	22-3104-28-07	MONO CAP 0.1UF +80-20%, 50V
	V	22-3104-28-06	MONO CAP 0.1UF +80-20%, 50V
C54,55		22-3223-25-72	MONO CAP 0.022UF +/-5%, 50V
	V	22-3223-25-36	MONO CAP 0.022UF +/-5%, 50V
C286,287,288		22-3680-26-00	CER CAP 68PF +/-10%
C195,198,265		22-6473-46-47	MYLAR CAP 0.047UF +/-10%, 100V
	A	22-6473-26-01	MYLAR CAP 0.047UF +/-10%, 50V
VC1		22-7002-01-00	TIIMMER CAP 4PF-20PF
	A	22-7002-00-00	TRIMMER CAP 20PF
D4,7-14,19,20		21-0001-00-00	DIODE 1N4148
	V	21-0001-00-02	DIODE 1N4148
D16,17		21-0032-00-00	DIODE 2-1K60
D2		21-0158-00-01	SCHOTTKY DIODE 1N5821 30V 3A
	V	21-0158-00-00	SCHOTTKY DIODE 1N5821 30V 3A
	A	21-0167-00-00	SCHOTTKY BARRIER RECTIFIER MBR3
	R	21-0133-00-00	SCHOTTKY RECTIFIER 1N5820 20V 3
Z1		21-0042-00-00	ZENER DIODE BZX79C5V1 5.1V 5MA
Z2		21-0039-02-00	ZENER DIODE MTZ13B 13V 10MA
	A	21-0039-01-00	ZENER DIODE BZX79B13 13V 500MW
Z7		21-0019-04-00	ZENER DIODE 6.8V MTZ6.8C IZ=20M
	A	21-0019-00-00	ZENER DIODE 6.8V 500MW 5MA
	A	21-0019-00-01	ZENER DIODE 6.8V 500MW 5MA
	A	21-0019-00-02	ZENER DIODE 6.8V 500MW 5MA BZX7
Z3-6		21-0006-05-00	ZENER DIODE MTZ5.6#B 5.45-5.73V
	A	21-0006-03-00	ZENER DIODE 5.6V 500MW 5MA BZX7
Q18		20-0011-02-01	TRANSISTOR 9018 H
Q7,8		20-0014-01-01	TRANSISTOR 9012 G
	A	20-0014-00-01	TRANSISTOR 9012 H
Q11.12		20-0019-00-01	TRANSISTOR 9014C
	V	20-0019-00-00	TRANSISTOR 9014C
	V	20-0019-00-05	TRANSISTOR H9014C
	A	20-0025-03-00	TRANSISTOR 1402D

DESTINATION		PART NUMBER	DESCRIPTION
Q17		20-0025-01-02	TRANSISTOR 1402B
	V	20-0025-01-01	TRANSISTOR ST 1402B
Q5,6,14,15		20-0025-03-00	TRANSISTOR 1402D
Q9,10		20-0003-02-00	TRANSISTOR 8050 C
	A	20-0028-02-00	TRANSISTOR NA31XJ
	A	20-0028-03-00	TRANSISTOR NA31XI/J
	A	20-0028-04-00	TRANSISTOR NA31X I/J/H
Q13,14		20-0033-01-00	TRANSISTOR ST1602D
Q3		20-0112-00-00	TRANSISTOR BD438 (SGS)
	A	20-0042-00-00	TRANSISTOR BD434
U29.30,31,32		20-0044-00-02	OPTO-COUPLER 4N27
	V	20-0044-00-00	OPTO-COUPLER 4N27
	V	20-0044-00-04	OPTO-COUPLER PC4N27
Q1,2		20-0105-00-00	POWER TRANSISTOR TIP42
	V	20-0105-00-01	POWER TRANSISTOR TIP42
	A	20-0105-01-00	PNP TRANSISTOR TIP42C
	A	20-0105-02-00	POWER TRANSISTOR TIP42A
	A	20-0105-03-00	POWER TRANSISTOR TIP42B

D.4. Schematics, PCB and Component layouts

Since the manufacturer is constantly improving the product, the circuits covered in this section may be different from that on the computer PCB without notice to the computer user. For detailed operation of the circuits, the user may refer to the relevant chapters of this manual.

D.4.1 Laser 128

The following descriptions are on the six pages of schematics of the Laser 128.

Page one (D-91)

a) Gate array 1

This gate array (U6) is for memory management, the parallel printer logic the mouse logic and the expansion slot signals.

b) The CPU (U20) is 65C02.

c) The CPU address buffers are U19 and U23. The CPU data bus buffer is U28.

d) The ROM is U15.

e) The keyboard controller (U21) is a 8048. U22 is for keyboard scanning decoding.

f) U29, U30, U31, U32 and U34 are for the paddle / mouse port interfacing.

g) XTAL1 and part of U26 is for the system clock (14.31818MHz) generation.

h) U18 and the associated circuits are used to generate the power-up reset and the CTRL - RESET reset.

i) U24 is the parallel port data latch.

Page two (D-92)

a) Gate array 3

This gate array (U38) contains circuits for the drive controllers, game port logic, the expansion RAM circuits, the ACIA logic, the keyboard logic and the sound output.

b) U27, part of U13 and U18 are for drive buffering.

c) U11 and U12 are the 6551 ACIA. U3 and U4 are RSC232 interfacing IC's.

Page three (D-93)

a) Gate array 2 (U25) is mainly for the video circuits.

b) U14 is the character generator ROM.

c) Part of U13 and U14 for video circuits.

d) U35, U36, U39 and U40 are the RAM chips.

e) U42 and U43 are the RAM data bus buffers.

f) U37 is the RAM for the disk drive controller use.

Page four (D-94)

This page contains information of the connectors. Some connectors pin assignments are found in other pages.

Page five (D-95)

This page is the switching mode power supply circuits for +5V and -12V. The +12V power supply regulator is also on this sheet.

Page six (D-96)

Page six is the crystal oscillator circuit for the ACIA.

D.4.2 Laser 128 EX

Page One (D-99)

a) Gate array 1

The gate array (U6) is for memory management, the parallel printer logic, the expansion slot signals and the mouse logic.

b) The CPU (U20) is 65C02-4.

c) The CPU address buffers are U19 and U23.

d) The ROM is U15.

e) The keyboard controller (U21) is a 8048. U22 is for keyboard scanning decoding.

f) U29, U30, U31, U32 and U34 are for the paddle / mouse port interfacing.

g) XTAL1 and part of U26 is for the system clock (14.31818MHz) generation.

h) U18 and the associated circuits are used to generate the power-up reset and the CTRL-RESET reset.

i) U24 is the parallel port data latch.

Page two (D-100)

a) Gate array 2 (U25) is mainly for the video circuits.

b) U14 is the character generator ROM.

c) U35 and U36 are the video RAM chips.

d) U37 is the RAM for the disk drive controller use.

e) U13 and U41 are for video circuits.

Page three (D-101)

a) The gate array three (U38) contains circuits for the drive controllers, game port logic, the expansion RAM logic circuits, the ACIA logic, the keyboard logic and the sound output.

b) U27 and part of U18 are for drive buffering.

c) U11 and U12 are the 6551 ACIA. U3 and U4 are RSC232 interfacing IC's.

d) U17 is the system RAM data bus buffer.

e) U7, U8, U9 and U10 are the system RAM.

Page four (D-102)

Page four contains information of the connectors. Some connectors pin assignments are found in other pages.

Page five (D-103)

a) U47 is the expansion RAM data bus buffer.

b) U45 and U46 are the expansion RAM. address buffers.

c) U48 to U79 are the expansion RAM chips.

Page six (D-104)

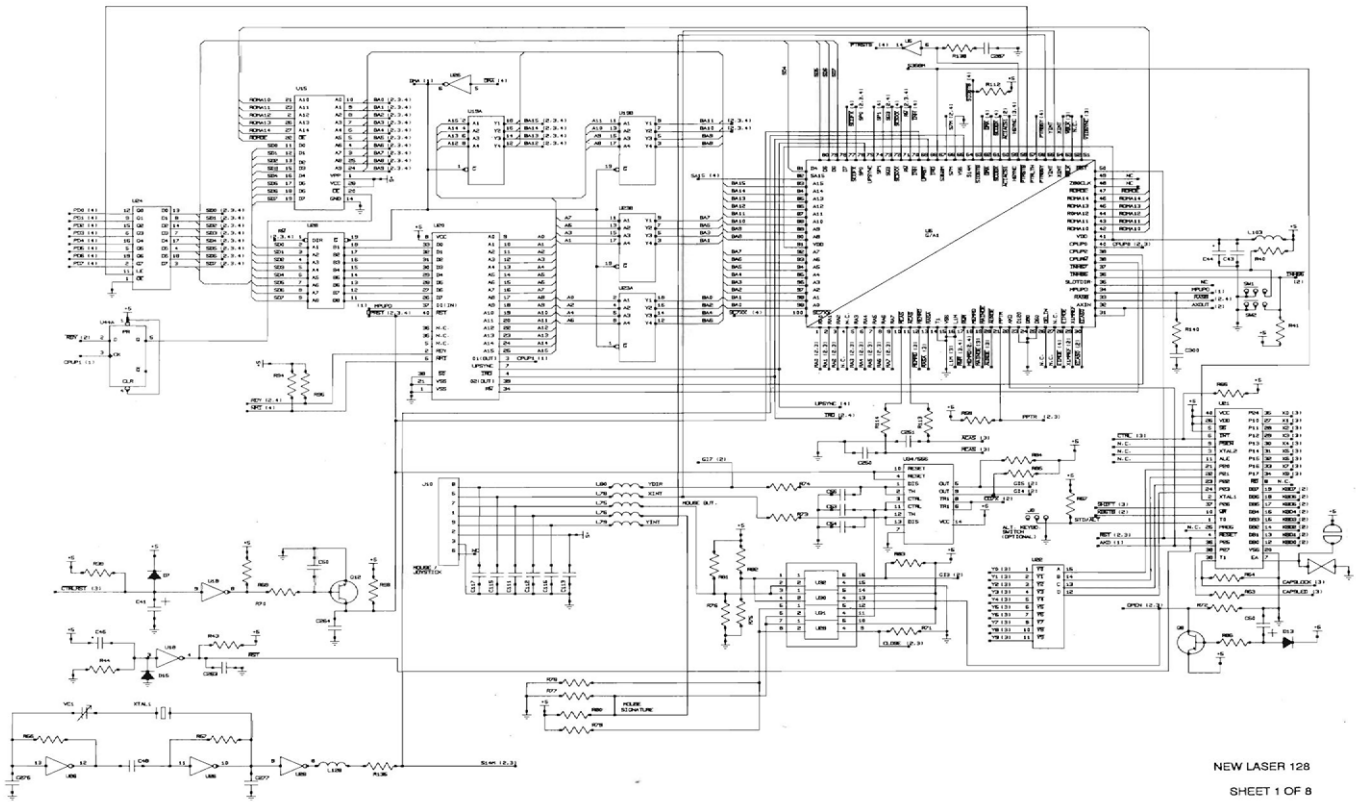
Page six is the switching mode power supply circuit for +5V and -12V. The +12V power supply regulator is also on this page.

Page seven (D-105)

Page seven is the crystal oscillator circuit for the ACIA.

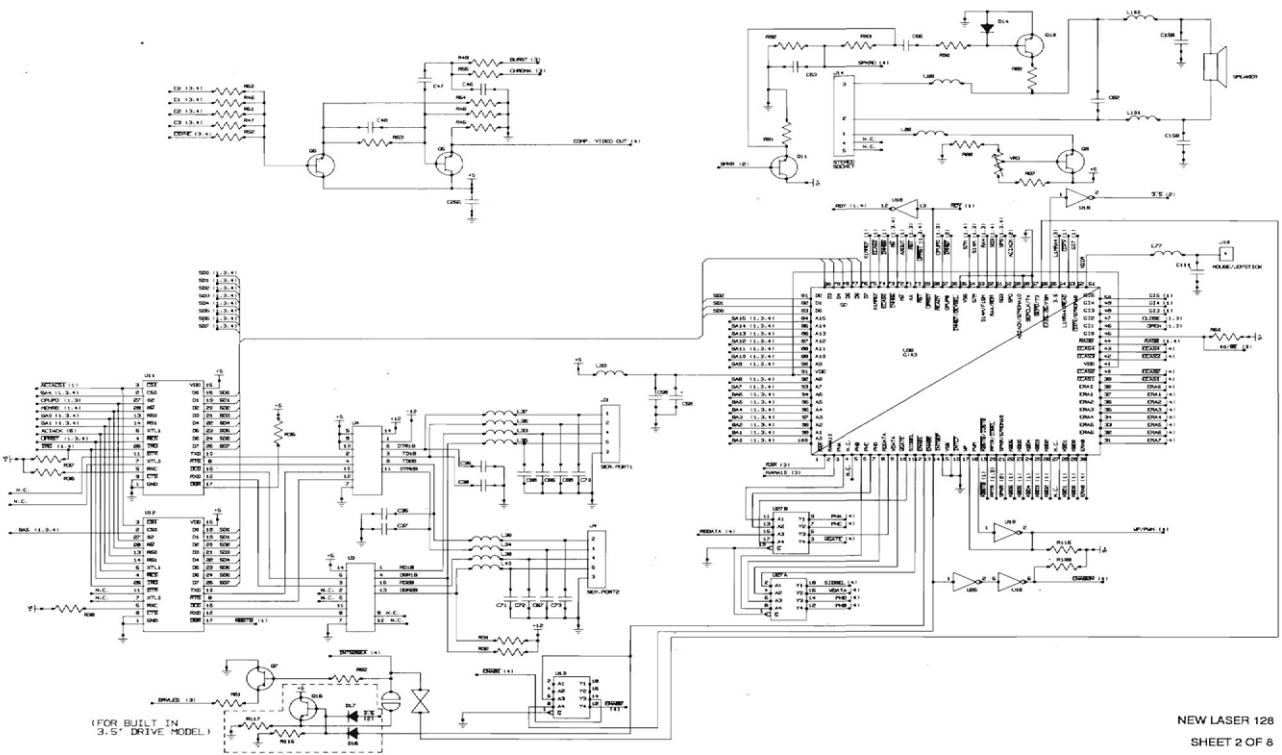
D.4.3

The schematic diagram shows the two expansion connectors and the expansion box power supply.

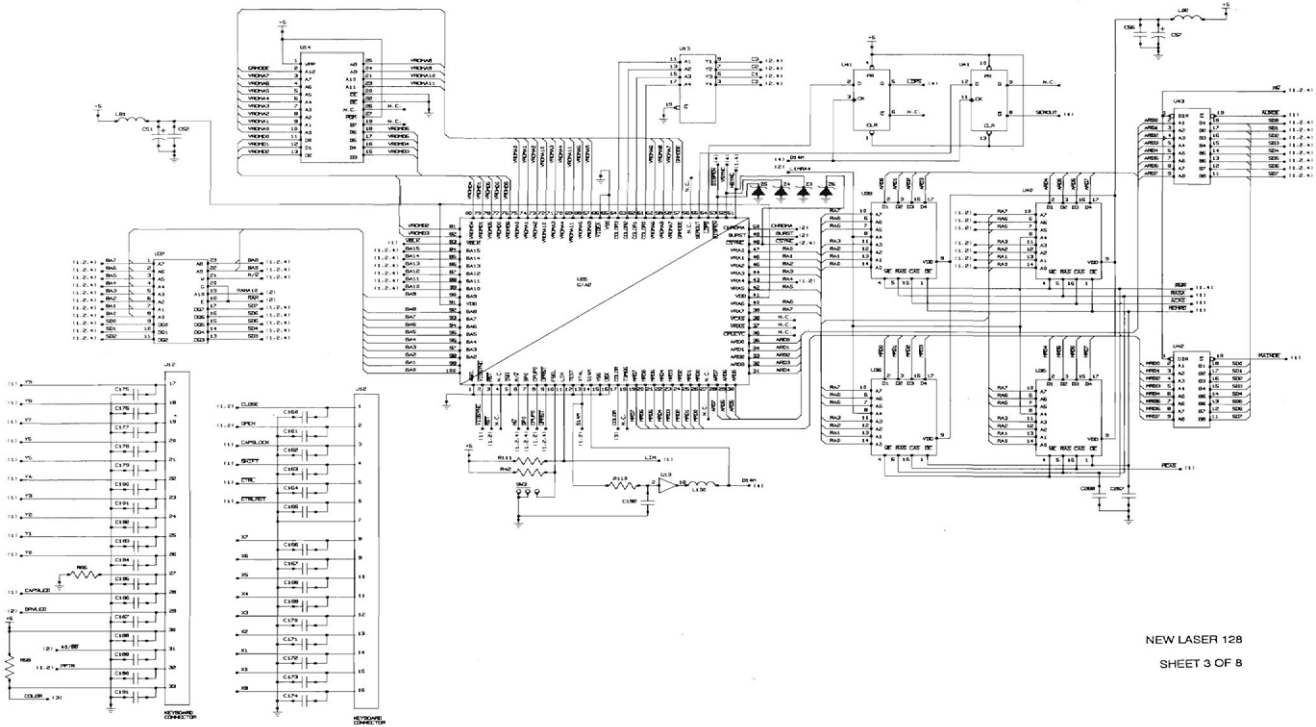


NEW LASER 128

SHEET 1 OF 8

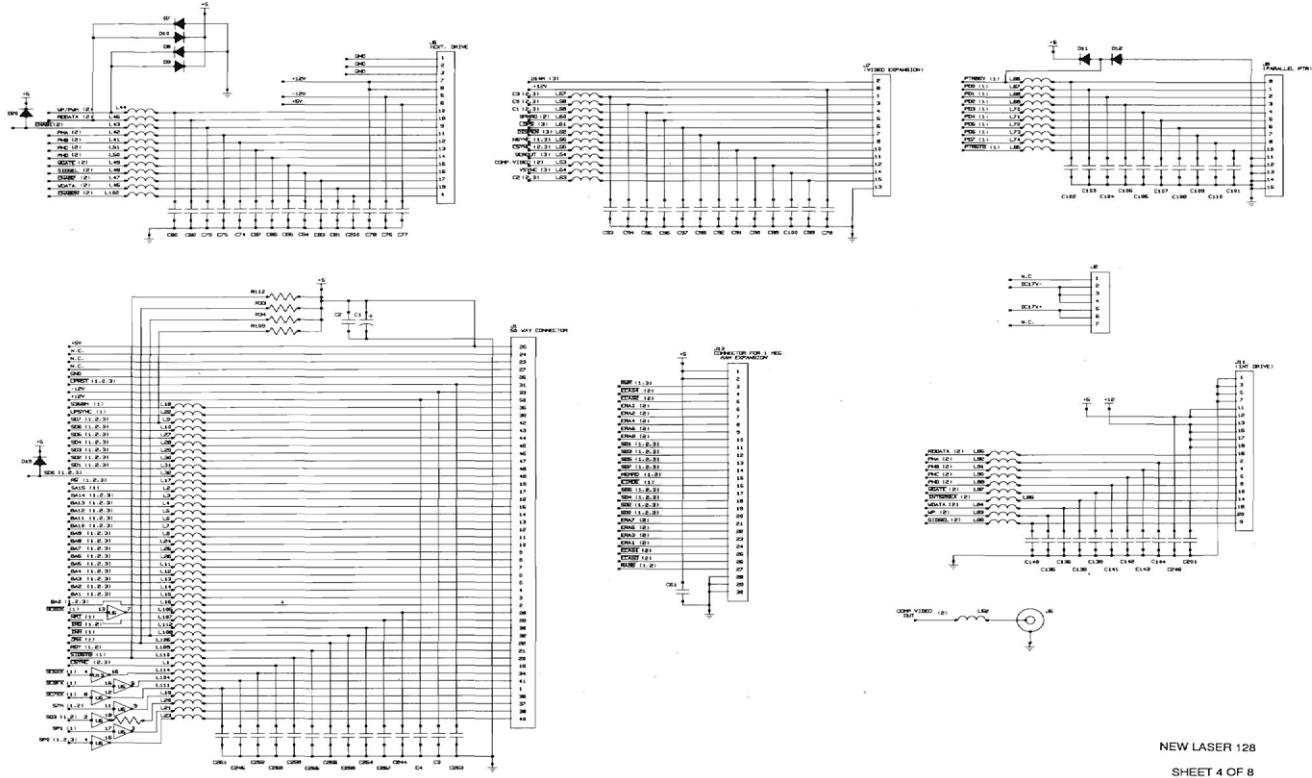


NEW LASER 128
SHEET 2 OF 8



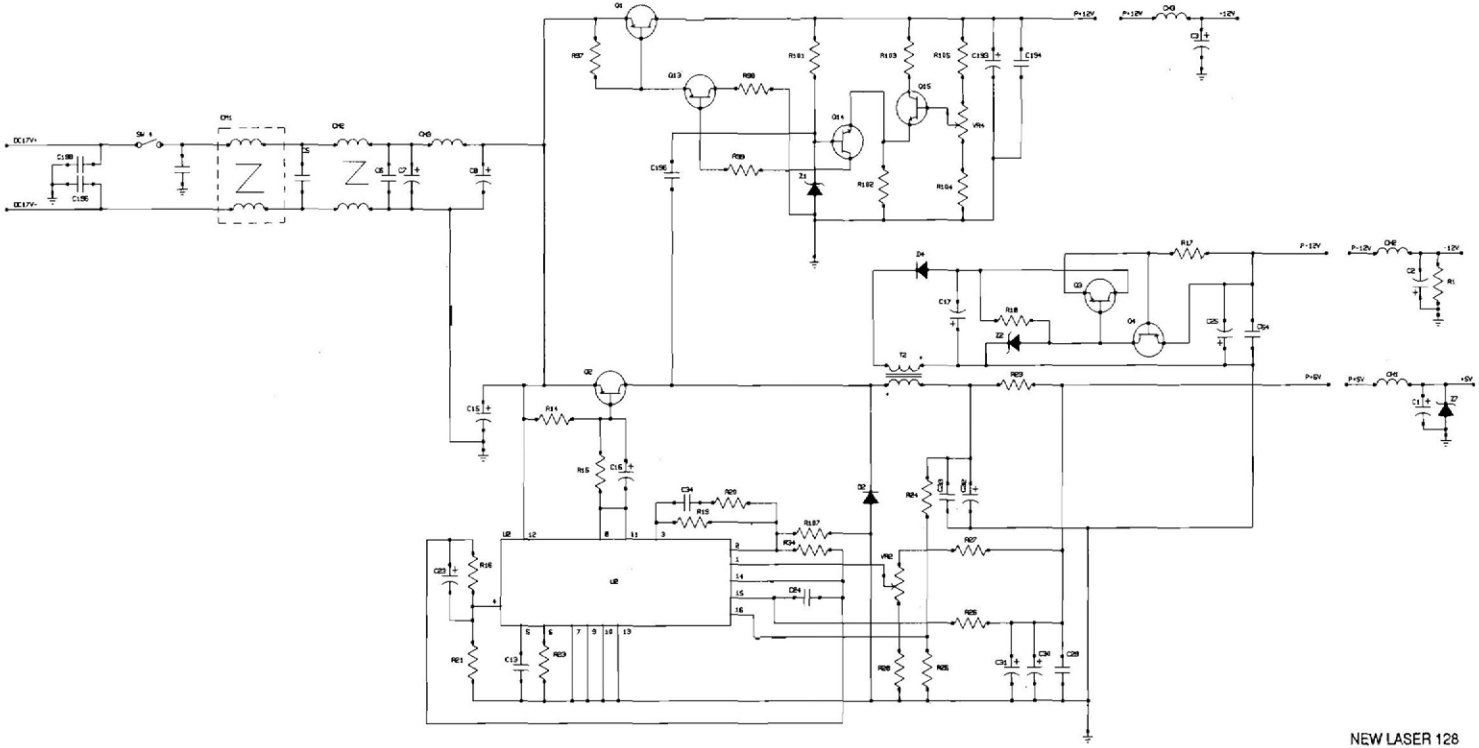
NEW LASER 128

SHEET 3 OF 8



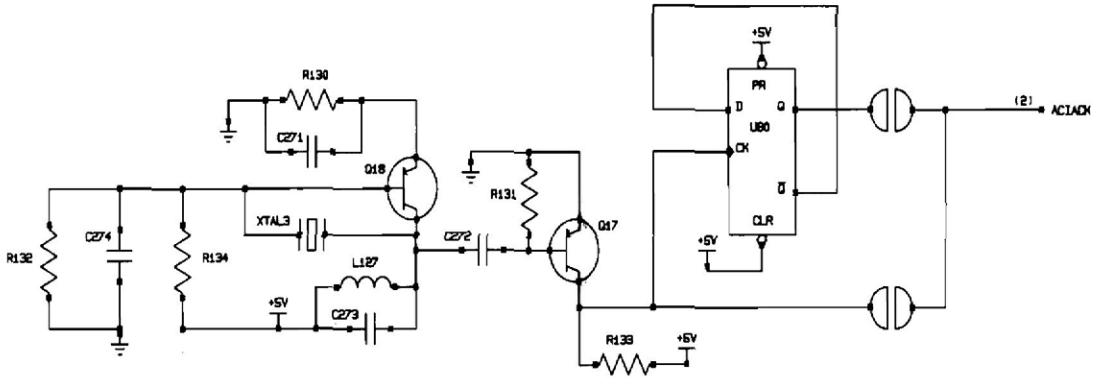
NEW LASER 128

SHEET 4 OF 8



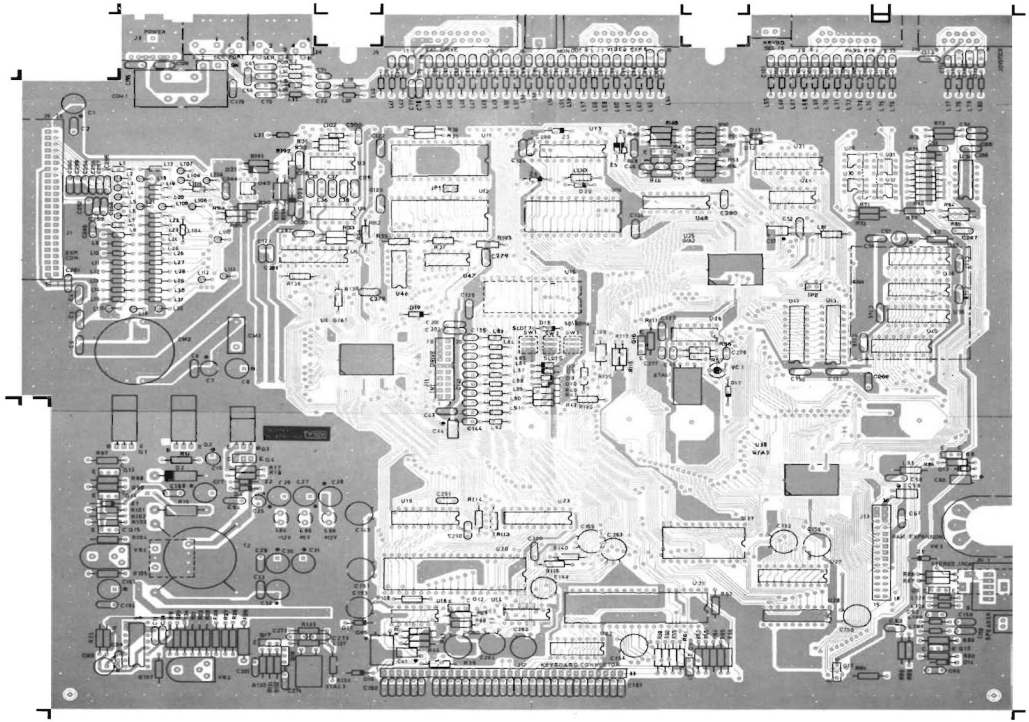
NEW LASER 128

SHEET 5 OF 8



NEW LASER 128

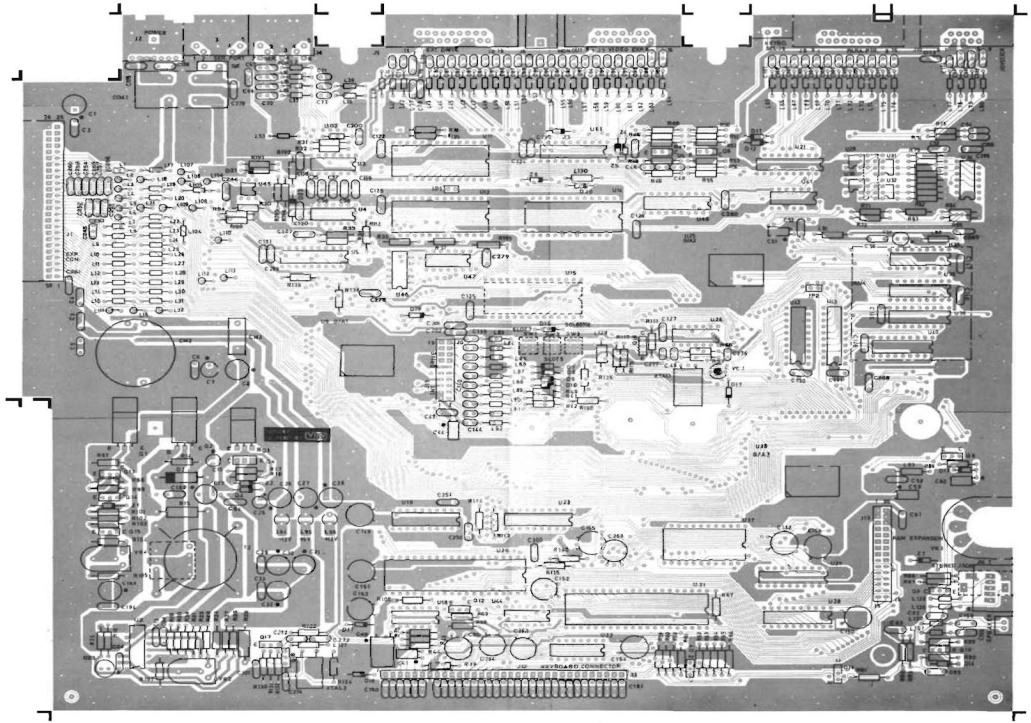
SHEET 6 OF 8



COMPONENT SIDE

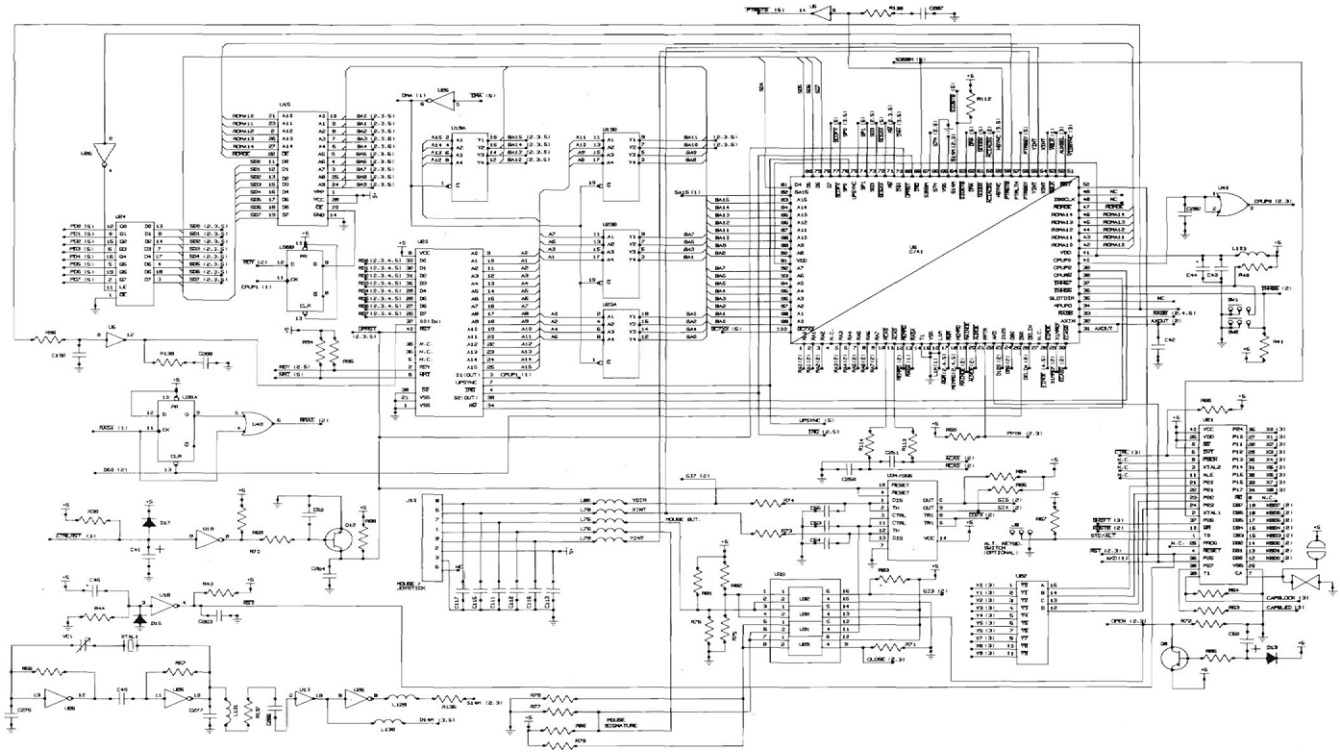
NEW LASER 128

SHEET 7 OF 8



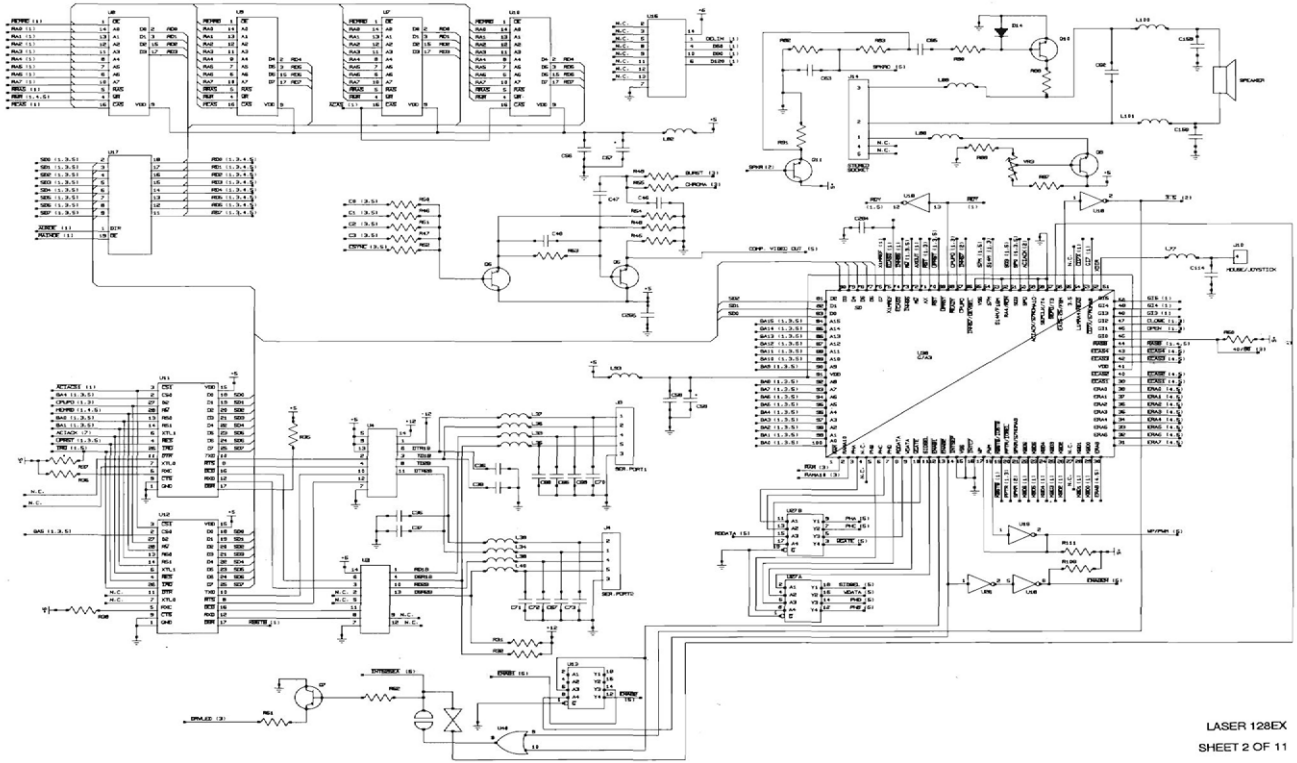
SOLDER SIDE

NEW LASER 128
SHEET 8 OF 8

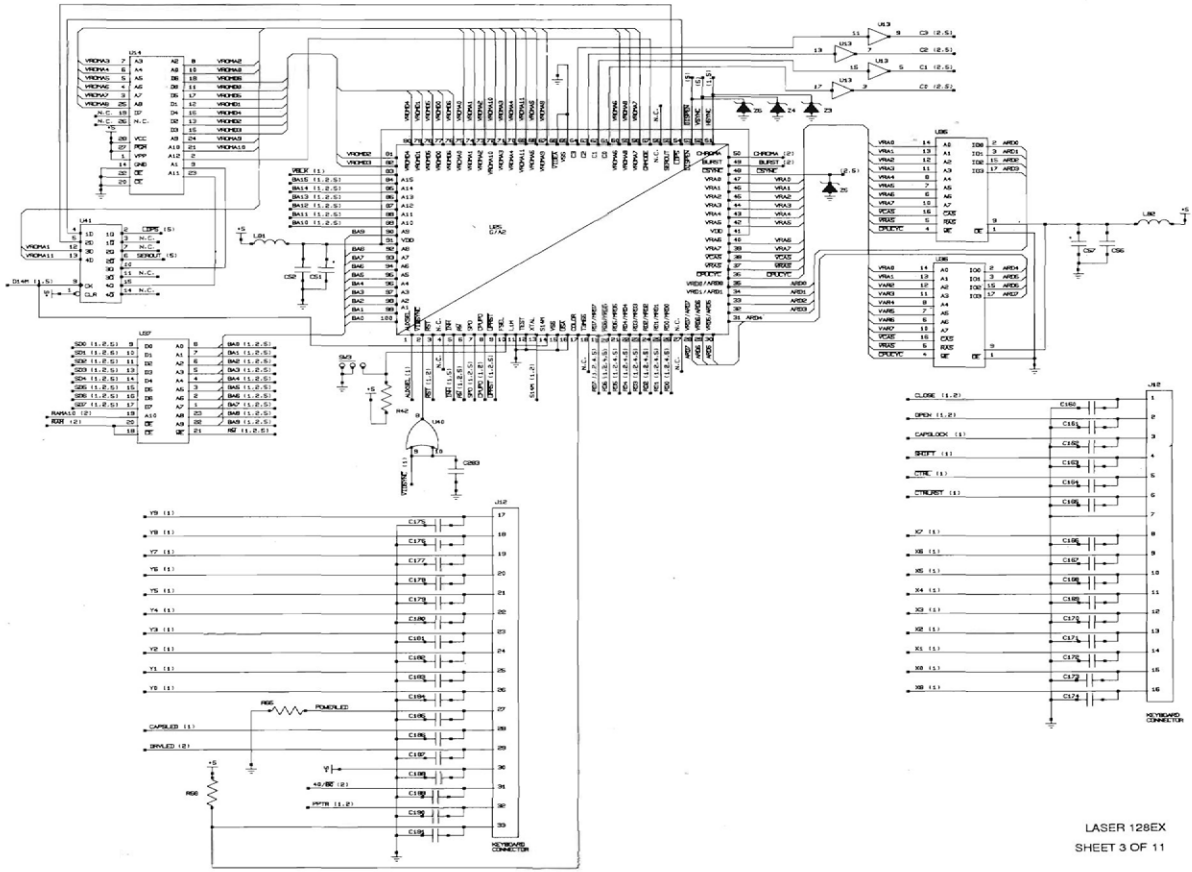


LASER 128EX

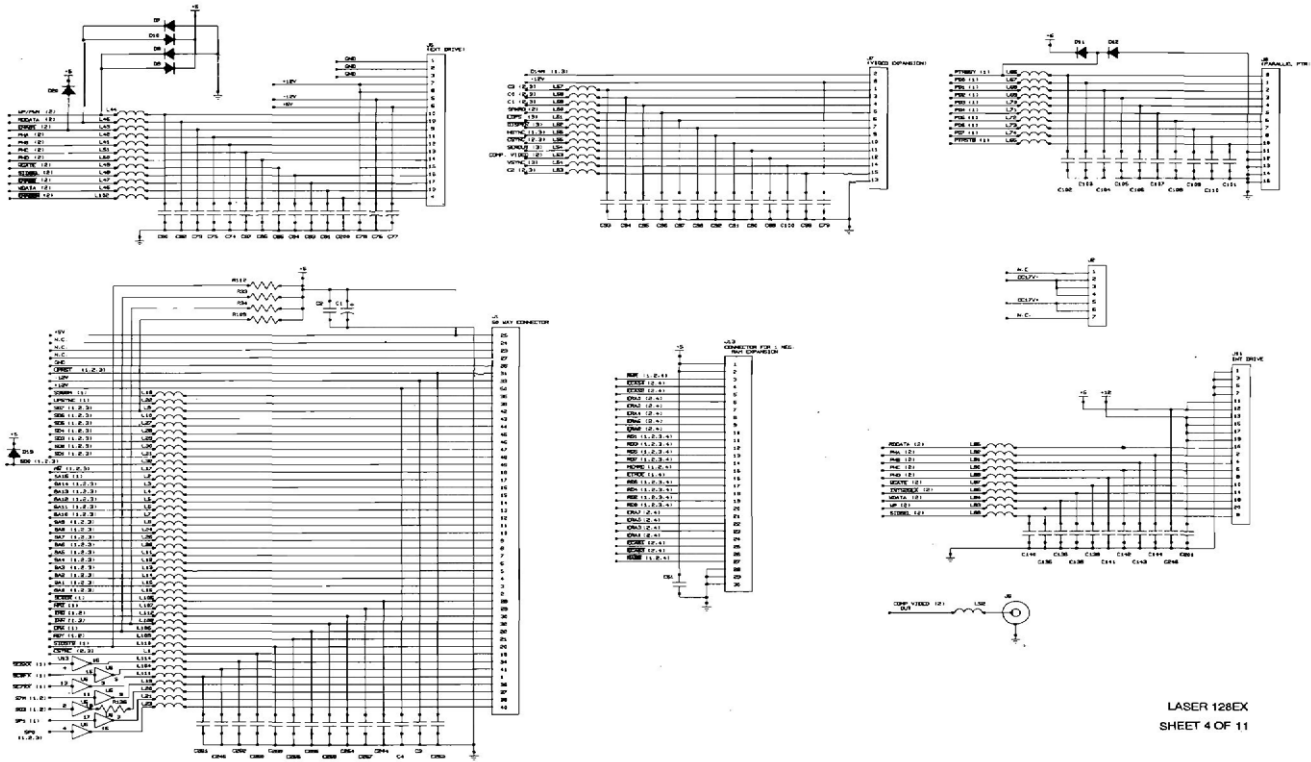
SHEET 1 OF 11



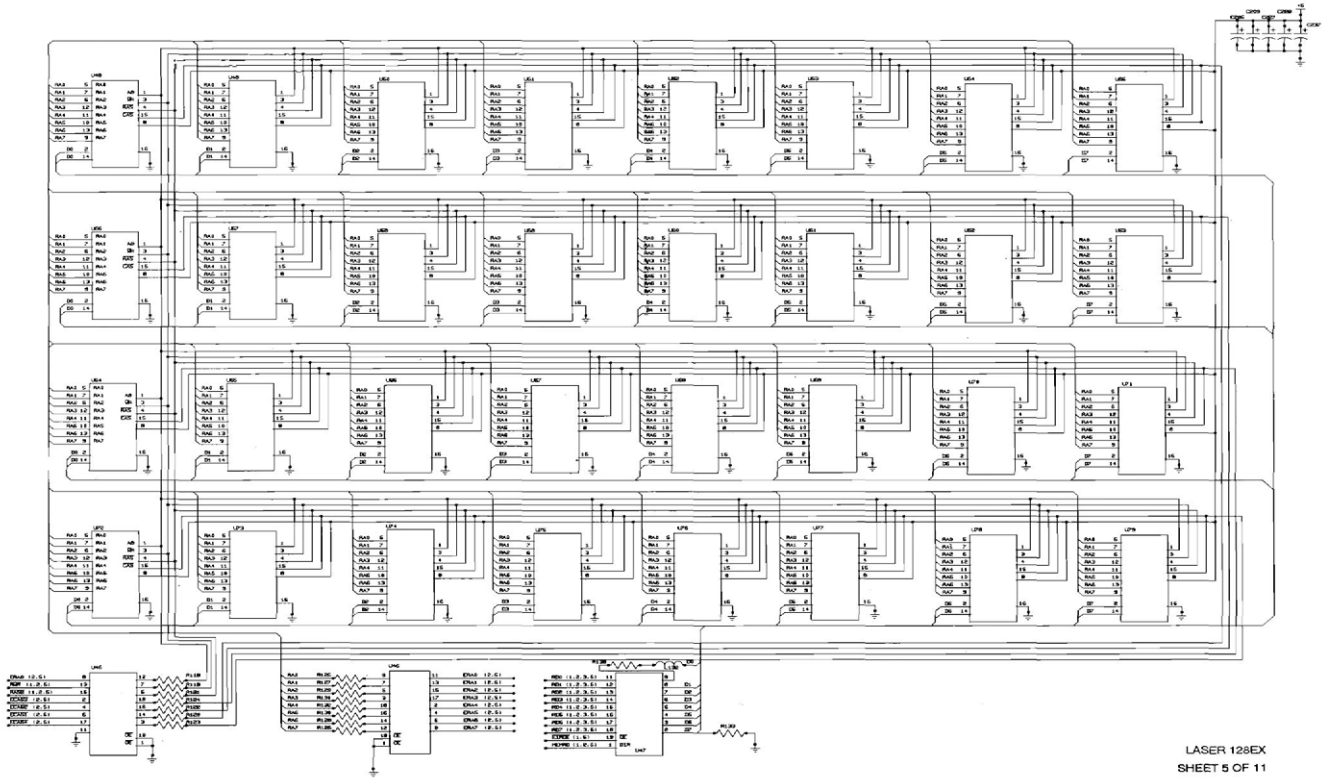
LASER 128EX
SHEET 2 OF 11



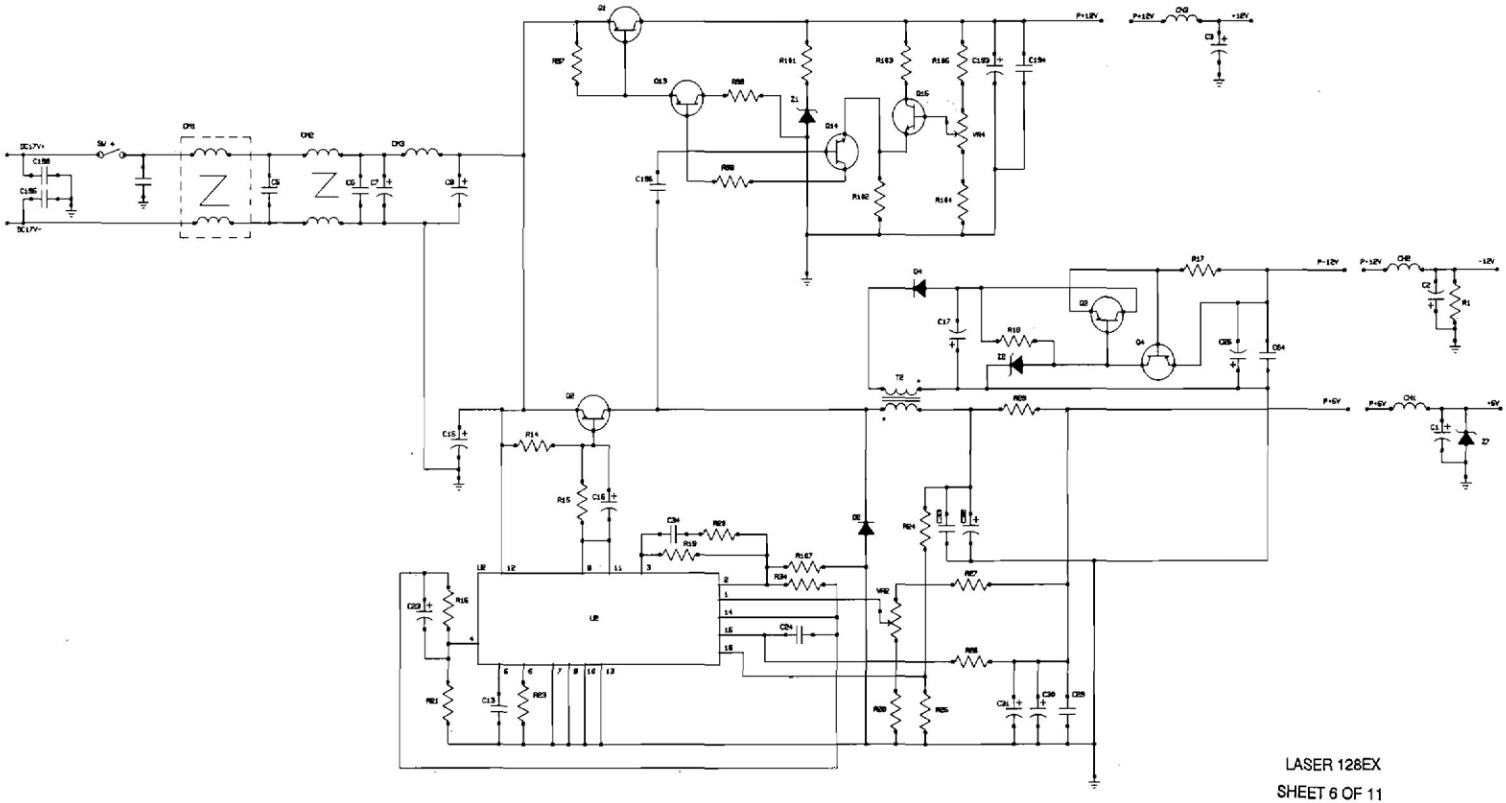
LASER 128EX
SHEET 3 OF 11



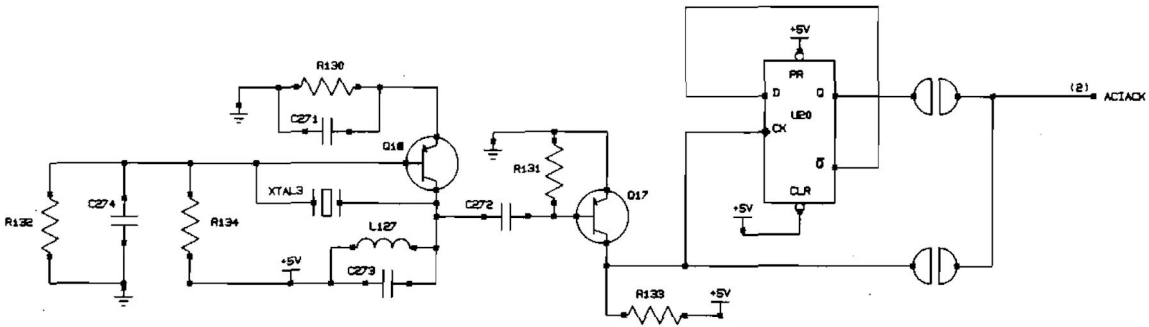
LASER 128EX
SHEET 4 OF 11



LASER 128EX
SHEET 5 OF 11

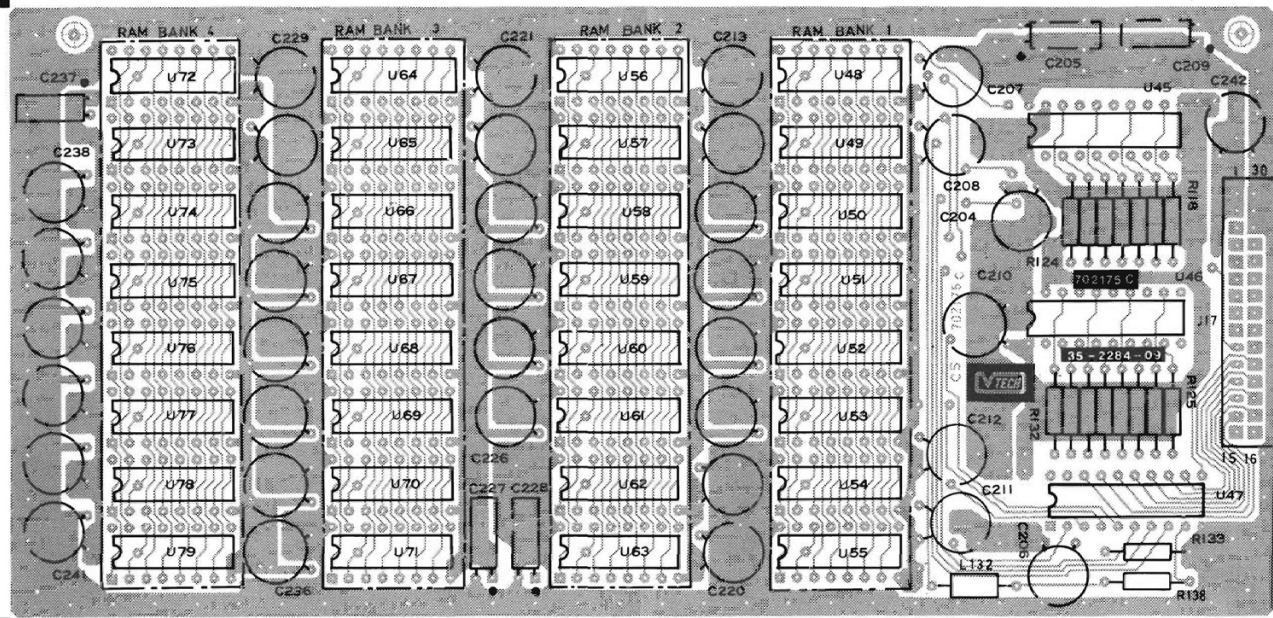


LASER 128EX
SHEET 6 OF 11



LASER 128EX

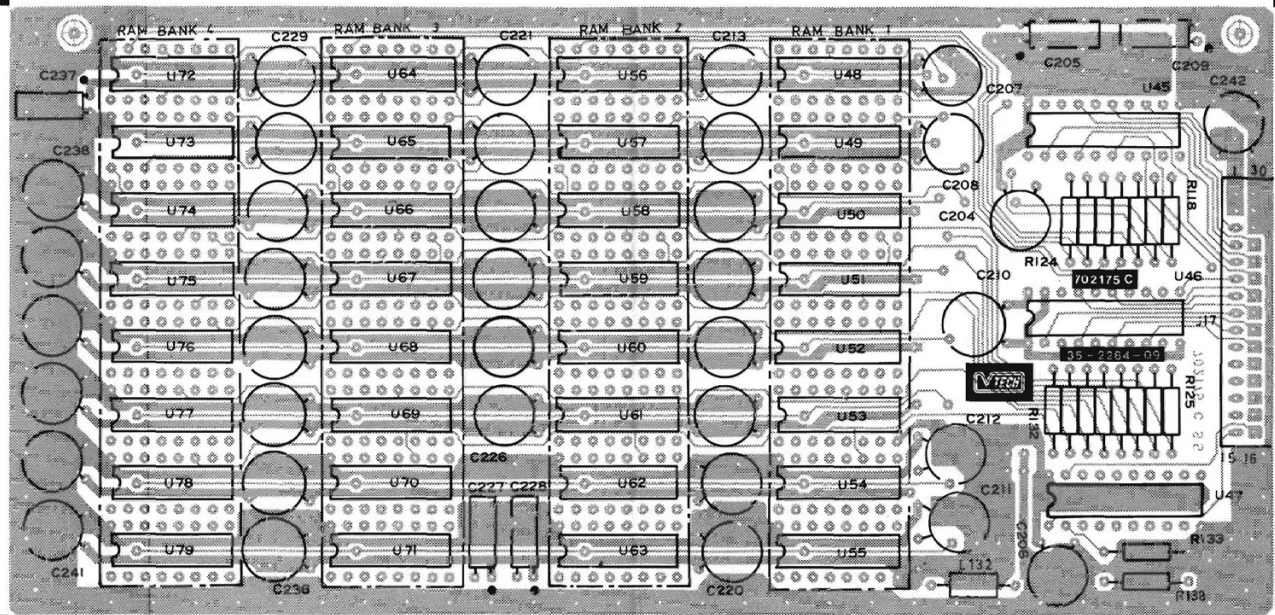
SHEET 7 OF 11



COMPONENT SIDE

LASER 128EX

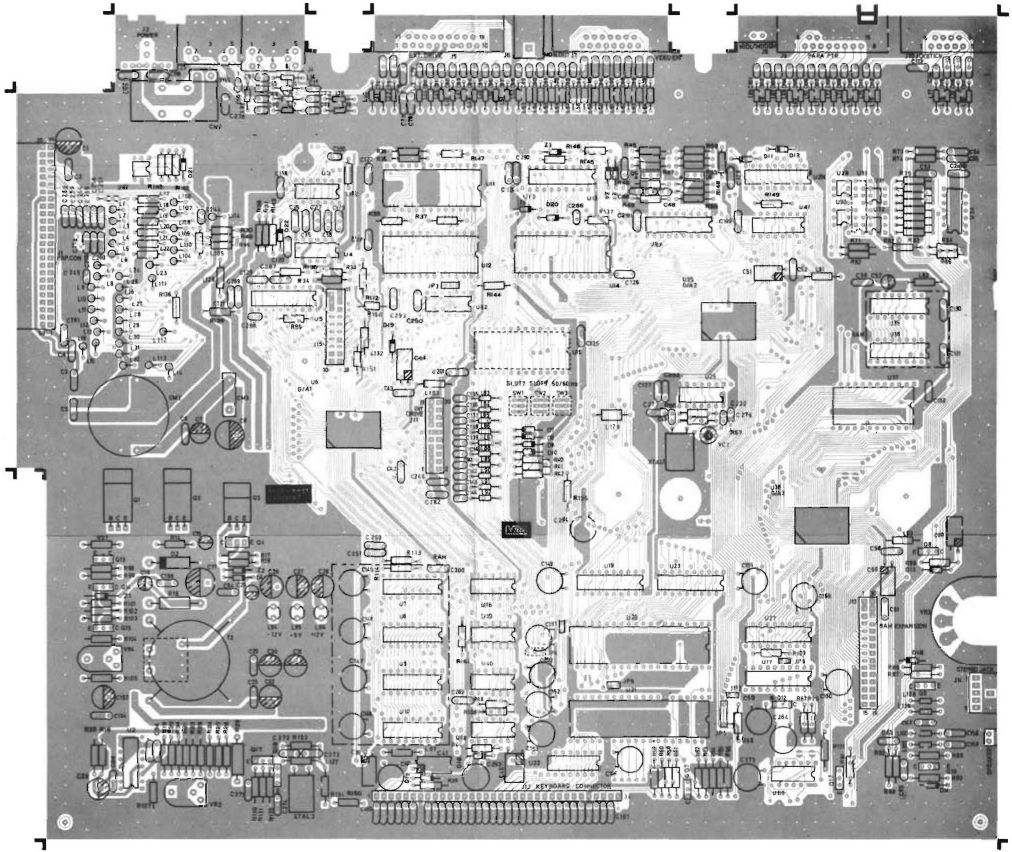
SHEET 8 OF 11



SOLDER SIDE

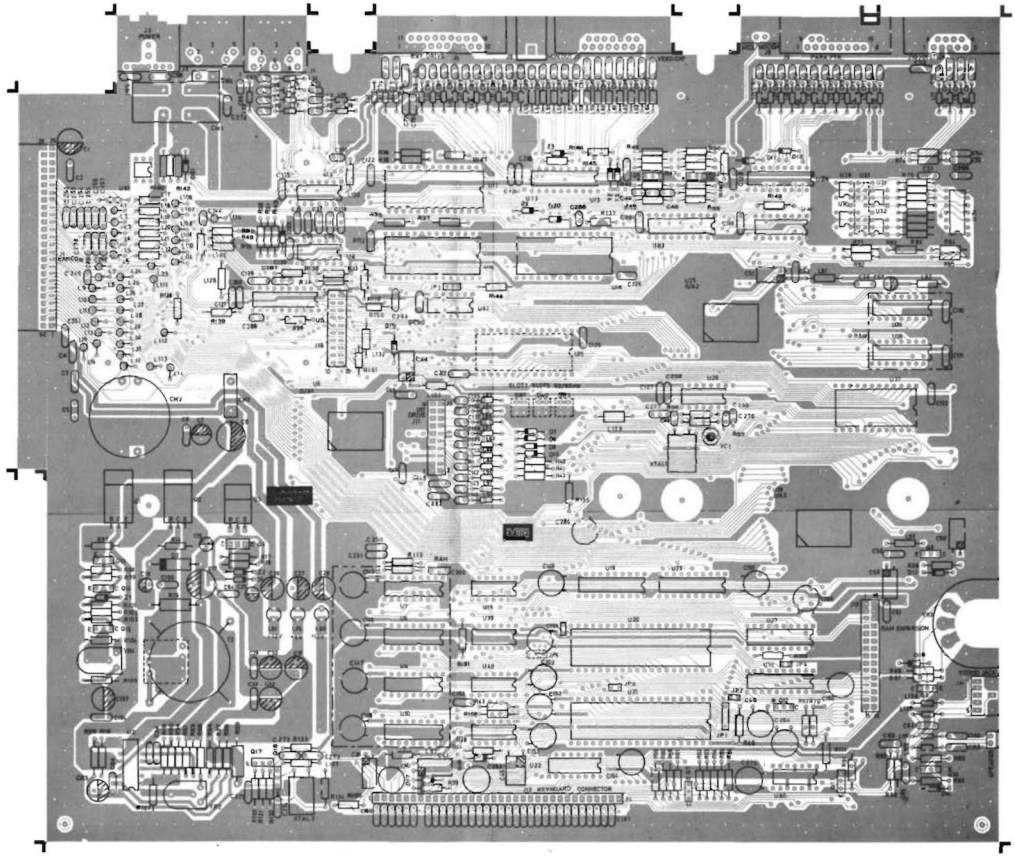
LASER 128EX

SHEET 9 OF 11



COMPONENT SIDE

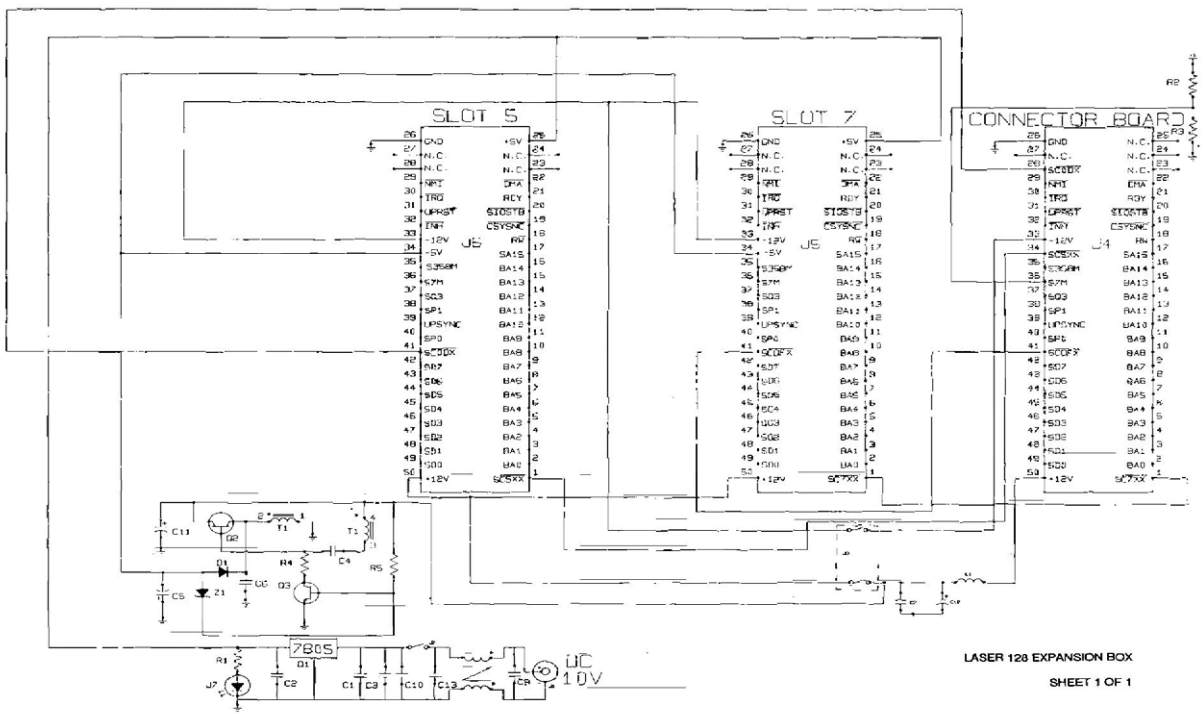
LASER 128EX
SHEET IO OF 11

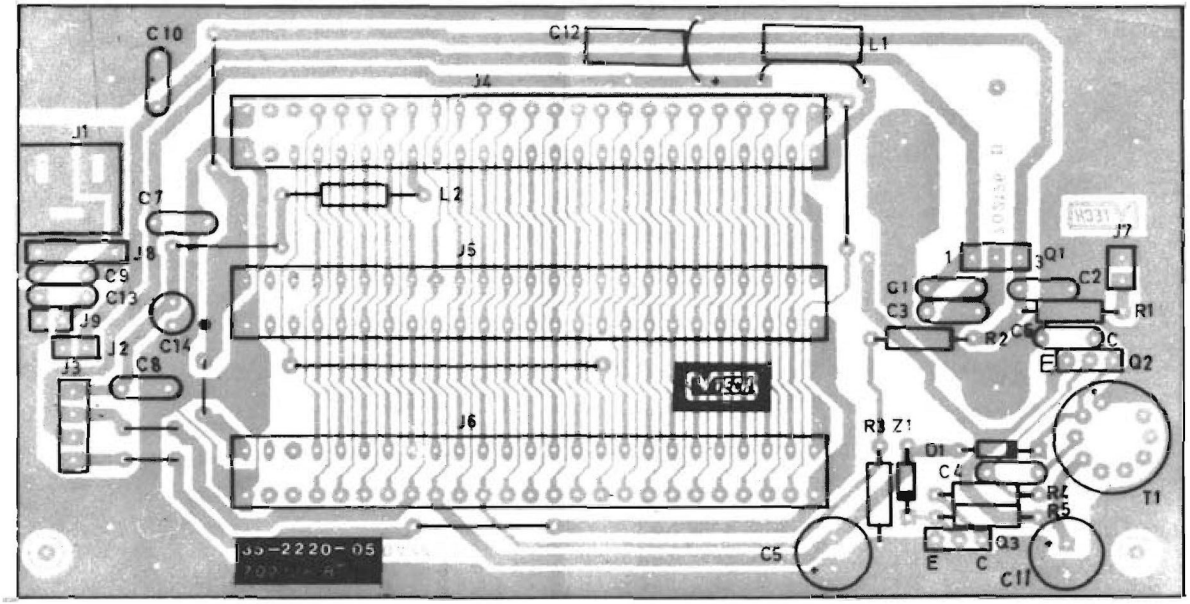


SOLDER SIDE

LASER 128EX

SHEET 11 OF 11


SLOT 5
SLOT 7
CONNECTOR BOARD
7805
LASER 128 EXPANSION BOX
SHEET 1 OF 1



LASER 128EX EXPANSION BD

SHEET 1 OF 1

91-2104-23
PRINTED IN HONG KONG