

OSBORNE 4

Theory of Operation Manual

-Revision 1.0-

Corresponds to Revision C Schematics

Osborne Computer Corporation

Technical Publications

August 1985

COPYRIGHT (C) 1985 OSBORNE COMPUTER CORPORATION
42680 Christy Street, Fremont, CA 94538
(415) 490-6885

OSBORNE 4 TECHNICAL SERIES
Theory of Operation Manual
Issue date: August 1985
Revision 1.0

Part Number: 2F40006-01

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopy, recording, or otherwise, without the prior written permission of OSBORNE COMPUTER CORPORATION.

The information in this document is subject to change without notice.

Neither OSBORNE COMPUTER CORPORATION nor this document makes any expressed or implied warranty, including, but not limited to the implied warranties of merchantability, quality, or fitness for a particular purpose. OSBORNE COMPUTER CORPORATION has no obligation to update or keep current the information contained in this document.

Under no circumstances will OSBORNE COMPUTER CORPORATION be liable for any loss or other damages arising out of the use of this manual

OSBORNE 4 is a registered trademark of OSBORNE COMPUTER CORPORATION.

**FEDERAL COMMUNICATIONS COMMISSION
RADIO FREQUENCY INTERFERENCE STATEMENT**

WARNING: This equipment has been certified to comply with the limits for a Class B computing device, pursuant to Subpart J of Part 15 of FCC rules. Only peripherals (computer input/output devices, terminals, printer, etc.) certified to comply with the Class B limits may be attached to this computer. Operation with non-certified peripherals is likely to result in interference to radio and TV reception.

INSTRUCTIONS TO USER: This equipment generates and uses radio frequency energy and if not installed properly, i.e., in strict accordance with the operating instructions, reference manuals, and the service manual, may cause interference to radio or television reception. It has been tested and found to comply with the limits for a Class B computing device pursuant to Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference when operated in a residential installation.

If this equipment does cause 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:

- o Reorient the receiving antenna.
- o Relocate the equipment with respect to the receiver.
- o Move the equipment away from the receiver.
- o Plug the equipment into a different outlet so that equipment and receiver are on different branch circuits.
- o Ensure that card mounting screws, attachment connector screws, and ground wires are tightly secured.
- o Ensure that card slot covers are in place when no card is installed.

If necessary, consult your dealer service representative for additional suggestions.

The manufacturer is not responsible for any radio or TV interference caused by unauthorized modifications to this equipment. It is the responsibility of the user to correct such interference.

CAUTION: This product is equipped with a UL listed and CSA certified plug for the user's safety. It is to be used in conjunction with a properly grounded 115 Vac receptacle to avoid electrical shock.

Table of Contents

Section 1 - General Information.....	1-1
1.0 Introduction.....	1-1
1.1 Description.....	1-1
1.2 Applicable Documents.....	1-2
Section 2 - System Timing.....	2-1
2.0 Introduction.....	2-1
2.1 Timebase (24 MegaHertz).....	2-1
2.2 Dot Clock (12 MegaHertz).....	2-1
2.3 CPU Clock (4 MegaHertz).....	2-2
2.4 USART Clock (2 MegaHertz).....	2-3
2.5 FDC Clock (1 MegaHertz).....	2-3
Section 3 - Central Processor Unit.....	3-1
3.0 Introduction.....	3-1
3.1 Z80A CPU.....	3-1
3.2 Address Drivers.....	3-5
3.3 Reset Logic.....	3-5
3.4 Interrupt Logic.....	3-6
3.4.1 Vertical Sync.....	3-8
3.4.2 FDC Command Completion.....	3-9
3.4.3 IEEE-488 Servicing.....	3-9
3.4.4 USART Servicing.....	3-10
3.4.5 RS-232-C Ring Indicator.....	3-11
Section 4 - Control/Decode Logic.....	4-1
4.0 Introduction.....	4-1
4.1 I/O Decode Logic.....	4-1
4.2 Control Logic.....	4-1
4.2.1 IORD/.....	4-2
4.2.2 IOWR/.....	4-2
Section 5 - RAM/ROM Address Control.....	5-1
5.0 Introduction.....	5-1
5.1 High Address Control Logic.....	5-1
5.2 ROM.....	5-3
5.3 Memory Map.....	5-4
Section 6 - Dynamic Random Access Memory.....	6-1
6.0 Introduction.....	6-1
6.1 Dynamic RAM Control Logic.....	6-1
6.2 Read Cycles.....	6-2
6.3 Write Cycles.....	6-3
6.4 Refresh Cycles.....	6-4

Table of Contents

Section 7 - Keyboard Control.....	7-1
7.0 Introduction.....	7-1
7.0.1 Key Decoding.....	7-1
7.0.2 Video Attributes.....	7-2
7.1 8155 PIO.....	7-4
7.1.1 8155 Overview.....	7-4
7.1.2 Command Register.....	7-4
7.1.3 I/O Addressing.....	7-4
7.1.4 Row Address.....	7-6
7.1.5 Column Address.....	7-6
7.1.6 Control Signals.....	7-6
Section 8 - Input/Output Board.....	8-1
8.0 Introduction.....	8-1
8.1 Parallel Interface.....	8-1
8.1.1 8155 PIO.....	8-1
8.1.2 8155 Overview.....	8-1
8.1.2.1 Command Register.....	8-2
8.1.2.2 I/O Addressing.....	8-2
8.1.2.3 Port A.....	8-3
8.1.2.4 Port B.....	8-4
8.1.2.5 Port C.....	8-5
8.1.2.6 Timer.....	8-5
8.1.3 IEEE-488 Interface.....	8-6
8.1.4 Centronics Interface.....	8-7
8.2 Serial Interface.....	8-8
8.2.1 USART Initialization.....	8-8
8.2.2 USART Servicing.....	8-9
8.2.3 RS-232-C Interface.....	8-9
Section 9 - Floppy Disk Control.....	9-1
9.0 Introduction.....	9-1
9.1 Z80A Interface.....	9-2
9.2 1797 FDC Operation.....	9-2
9.2.1 FDC Commands.....	9-3
9.2.1.1 Type I Commands.....	9-3
9.2.1.2 Type II Commands.....	9-4
9.2.1.3 Type III Commands.....	9-4
9.2.1.4 Type IV Commands.....	9-4
9.2.2 FDC Status.....	9-5
9.2.3 FDC Read Operation.....	9-5
9.2.4 FDC Write Operation.....	9-5
9.3 Floppy Disk Drive Interface.....	9-5
9.4 Write Precompensation.....	9-7
9.5 Data Separation.....	9-8

Table of Contents

Section 10 - Display Control.....	10-1
10.0 Introduction.....	10-1
10.0.1 Character Address Counter.....	10-4
10.0.2 Scan Counter.....	10-4
10.0.3 Text Address Counter.....	10-4
10.1 Display Timing Control.....	10-4
10.2 Address Control.....	10-5
10.2.1 Column Counter.....	10-5
10.2.2 Scan Address Counter.....	10-5
10.2.3 Row Address Counter.....	10-6
10.3 Control ROM.....	10-6
10.4 Line Buffer.....	10-8
10.4.1 Line Buffer Control.....	10-9
10.4.2 Writing to the Line Buffer.....	10-11
10.4.3 Reading from the Line Buffer.....	10-13
10.5 Character Buffer.....	10-13
10.6 Character Generator ROM.....	10-13
10.7 Output Video Control.....	10-18
10.7.1 Parallel to Serial Converter.....	10-18
10.7.2 Reverse Video.....	10-18
10.7.3 Alternate Character Set.....	10-19
10.7.4 Video Monitor Control.....	10-20
10.7.4.1 Video Out.....	10-20
10.7.4.2 Horizontal Sync.....	10-21
10.7.5 Composite Video Output.....	10-21

Figures

Figure 1-1	Detailed Block Diagram.....	Foldout
Figure 3-1	Basic Z80A CPU Timing.....	3-2
Figure 5-1	Osborne 4 Memory Map.....	5-5
Figure 7-1	Row and Column Keyboard Addresses.....	7-3
Figure 10-1	Video Display Character Organization.....	10-1
Figure 10-2	Memory Map of Video Data.....	10-1
Figure 10-3	RAM Video Memory Address/Screen Position.....	10-2
Figure 10-4	Management of Line Buffer/RAM Segments.....	10-8
Figure 10-5	RAM Address During Refresh Cycle.....	10-12
Figure 10-6	128 Character Set ROM Organization.....	10-15
Figure 10-7	256 Character Set ROM Organization.....	10-16
Figure 10-8	Hexadecimal to ASCII Conversion.....	10-17

Tables

Table 3-1	Z80A Pin Functions.....	3-3
Table 3-2	Interrupt Enable Flip-Flop Control.....	3-9
Table 4-1	I/O Port Addresses.....	4-3
Table 5-1	RAM/ROM Selection Criteria.....	5-3
Table 7-1	System Board 8155 Addressing.....	7-5
Table 8-1	I/O Board 8155 Addressing.....	8-3
Table 9-1	Diskette Capacities.....	9-1
Table 9-2	FDC Internal Register Addressing.....	9-2
Table 10-1	Control ROM (3J) Timing and Control.....	10-7
Table 10-2	Line Buffer RAM Segment Read/Write Control....	10-13
Table 10-3	Video Output Control Options.....	10-20

1.0 Introduction

This document provides detailed information about the Osborne 4 portable computer. The explanation ranges from an overall discussion of system fundamentals to a gate-level analysis of individual circuits.

By referring to the block diagram on the following page, you can select the section describing a specific circuit. The Table of Contents then provides the page numbers corresponding to the selected sections.

Each section is written with an introductory paragraph summarizing the general content of the section to save you time in trying to locate specific details of circuit operation.

A system of zone coordinates will be used to locate circuits on the schematics in the descriptions. Zone D4 of sheet 1, for example, would refer to the upper left corner of schematic sheet number one.

1.1 Description

The Osborne 4 is a 22-pound, self-contained computer which includes disk drives, keyboard, CRT screen, and software. The material in this document focuses on the two electronics boards housed within the Osborne 4, the System Board and the I/O Board. The components included on each board are listed below:

System Board

- o System Timing Network
- o 4 MHz Z80A CPU
- o Keyboard Control
- o Floppy Disk Control
- o 64 Kbyte Dynamic RAM
- o 4 Kbyte EPROM
- o Display Controller

I/O Board

- o Centronics Parallel I/O
- o RS-232-C Serial I/O

Detailed information on each of the areas mentioned above is available in the appropriate section of this document.

1.0 -- General Information

1.2 Applicable Documents

1. Intel Component Data Catalog, January 1982
2. Western Digital Corp. 1797 FDC Data Sheets
3. Zilog Corp. Z80A Data Sheets
4. EIA Standard RS-232-C, August 1969
5. IEEE Standard 488-1975
6. Centronics Parallel Interface Standard
7. 8251A USART Data Sheets
8. 8155 PIO Data Sheets

2.0 Introduction

This section describes the generation and purpose of the clock signals provided on the Osborne 4. The timing relationships between the various signals are also discussed.

2.1 Timebase (24 MegaHertz)

The 24 MHz main clock is generated by a crystal-controlled RC (resistor/capacitor) oscillator. The actual output of the oscillator is 23.9616 MHz, shortened to 24 MHz for brevity. This circuit is located at zone D4 on sheet 1. The 24 MHz signal, available at integrated circuit package 5H pin 6 and at test point TP7, is used to drive other logic to produce the following timing:

- o 12 MHz Dot Clock for the display
- o 4 MHz CPU Clock for the Z80A
- o 2 MHz USART Clock
- o 1 MHz FDC Clock

2.2 Dot Clock (12 MegaHertz)

The 12 MHz Dot Clock is generated through a division of the 24 MHz timebase. This division is done by gate 6J and D flip-flops 5J, located at zone D2 and C3 of sheet 1. The 24 MHz clock, the 12 MHz Dot clock, and the 4 MHz CPU clock are all generated in a definite and fixed phase relationship with respect to each other.

Since all 4 D flip-flops contained in 5J may be in indeterminate states after power-up, and, in addition, the phase relationship may change due to noise or transients, a circuit is provided to clear all four flip-flops to a known state that will cause the phase relationship to be resynchronized. This circuit provides a reliable method of maintaining the strict timing relationships required in the Osborne 4. The 4-input NAND-gate 4H provides a clear pulse to all four flip-flops should the phase relationship become incorrect. When the clock circuitry is functioning correctly, and the proper phase relationship is being maintained, no pulse will be generated at 5J pin 1.

If synchronization is lost by the 4 D flip flops, a 4 MHz active-low pulse with the duration of a single 24 MHz clock cycle is generated at 6J pin 11. This pulse is inverted by the next stage of 6J and presented to the D input of 5J at pin 4. On the next rising edge of the 24 MHz clock, the flip-flop output goes high. At this moment, both pins 4

2.0 System Timing

and 5 of 6J are now high, forcing pin 6 low. On the next 24 MHz clock rising edge, the flip-flop output goes low, and causes the input of flip-flop 5J to go high again through the inversion of 6J. In this manner, a 50% duty cycle 12-MHz Dot clock is generated at 5J pin 2. The role of the active-low pulse provided at 6J pin 11 ensures the proper synchronization of the Dot clock with the 24 MHz and 4 MHz clocks.

The 12MHz Dot clock serves two basic purposes:

1. Generation of video display control timing signals. (See sheet 7, zone B4)
2. Serial shift control of the individual dots that compose each scanned row of a character on the screen (See sheet 8, zone C4)

2.3 CPU Clock (4 MegaHertz)

Three D flip-flops in zones D2 and D3 of sheet one generate the 4 MHz CPU clock. A common 24 MHz clock drives all three flip-flops. To analyze how these flip-flops function, assume that all three start off cleared and with the Q outputs of each flip-flop at a low level. On the first rising edge of the 24 MHz clock, the pin 10 output of 5J goes high since its D input is at a high level as provided by 5J pin 14. On the clock's next rising edge, the pin 7 output of 5J goes high because the pin 5 D input is high. Finally, on the 24 MHz clock's third rising edge, the pin 15 output goes high, since its D input is high. At this time, the pin 14 output changes from high to low. The low-level signal at pin 14 is connected to pin 12 and the next sequence of three clocks resets each of the flip-flops, one at a time in turn.

Thus, the first three clock cycles sequentially set the flip-flops, and the next three clock cycles sequentially reset the flip-flops. This takes a full six clock cycles to complete the sequence. 24 MHz divided by six yields 4 MHz. This represents the 4MHz clock signal generated at 5J pin 14 which regulates the timing of the Z80A CPU.

The 4 MHz CPU clock signal is used for the following purposes:

- o Z80A CPU Clock
- o Used in generating USART and FDC clocks
- o Used in generating Column Address Select (CAS/)
- o 8155 PIO Clock
- o Used in generating IORD/, IOWR/, and WR/
- o FDC Data Separator Clock

2.0 System Timing

The 4 MHz clock signal input to the Z80A must be treated with special care. For 4 MHz Z80A operation, the sum of the clock rise and fall times must not exceed 30 nanoseconds. Resistor R14, a 390 ohm resistor is pulled up to +12 volts, and diodes CR1 and CR2 are connected to the Z80A clock input line. These germanium diodes have a forward voltage drop of 0.3 volts; resulting in a bias voltage to the clock line of +5.3 volts. This meets the Z80A specification for maximum clock input voltage. The 74LS04 drivers contained in 5H have parallel outputs and can sink sufficient current together to meet the fall time requirement of the clock. However, the 74LS04 gates cannot provide sufficient source current by themselves to meet the risetime requirement. This is why the clock line is tied to +12 volts through R14. When the outputs of the two 74LS04 gates switch to a high level, the +12 volt supply provides an additional source of current to rapidly charge up the line in time to meet the 30 nanosecond risetime required by the CPU clock input. The diodes prevent the clock line from rising to the full +12 volt level and clamp it to +5.3 volts.

2.4 USART Clock (2 MegaHertz)

The 8251A USART is located on the I/O board and provides serial I/O capability. This USART must be provided with a clock for internal device timing. The clock must, by manufacturer's specification, run at a speed greater than 30 times the highest serial data rate of the USART. The 2 MHz clock easily provides that rate.

2.5 FDC Clock (1 MegaHertz)

The 1797 floppy disk controller requires a free-running squarewave clock for its internal timing. For the 1797 to control the 5-1/4 inch floppy diskette drives used in the Osborne 4, the clock must have a frequency of 1 MHz.

3.0 Introduction

This section describes the Z80A CPU, and its associated address drivers, reset logic, and interrupt logic. Fundamentals of the Z80A application for the Osborne 4 are covered; more specific details can be obtained from the Z80A IC Specifications.

3.1 Z80A CPU

The Z80A is the heart of the Osborne 4 portable computer. Its function is to obtain instructions from memory (RAM or ROM) and perform the desired operations. The memory contains instructions and data to be processed. As each instruction is fetched from memory, it is placed in the instruction register and decoded. Each instruction is unique, and when decoded causes a specific function to be performed. The Z80A control section does the decoding and then generates all of the control signals needed to read or write data to or from the CPU internal registers or external memory.

The Z80A contains eighteen 8-bit registers and four 16-bit registers. These registers are organized as follows:

Eight Bit Registers

Accumulator	2
Flags	2
General Purpose	12
Interrupt Vector	1
Memory Refresh	1
	18

Sixteen Bit Registers

Index (IX)	1
Index (IY)	1
Stack Pointer	1
Program Counter	1
	4

The 12 general-purpose 8-bit registers may be used individually or in pairs, as 16-bit registers.

3.0 -- Central Processor Unit

The Z80A CPU executes instructions by stepping through a very precise set of a few basic operations including:

- o Memory Read or Write
- o I/O Device Read or Write
- o Interrupt Acknowledge

All instructions consist of a series of these three operations. Each of the basic operations can take from three to six clock periods. The basic clock period is known as a T state, and the basic operations are known as machine cycles. Figure 3-1 shows the execution sequence for a typical instruction. This particular instruction consists of three machine cycles: M1, M2, and M3. The first machine cycle of any instruction, M1, is always the fetch cycle. It can be four, five, or six T states long, depending on the instruction. M1 is used to read in from memory the opcode of the next instruction to be executed. The following machine cycles transfer data between the Z80A and memory or I/O devices as needed to complete the action specified by the opcode. These machine cycles may consist of from three to five T states.

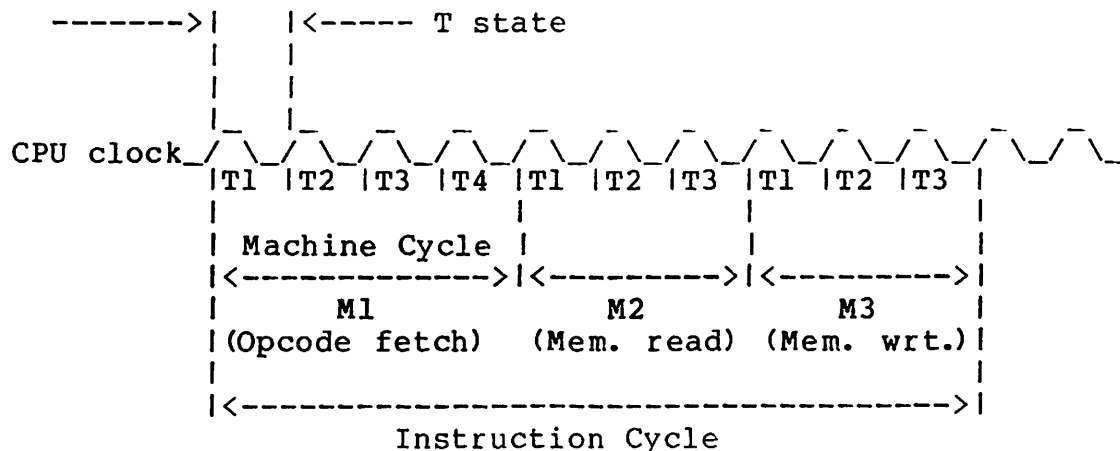


Figure 3-1. Basic Z80A CPU Timing

The Z80A is housed in an industry-standard 40-pin DIP package. Table 3-1 lists the pins and their associated functions.

Table 3-1. Z80A Pin Functions

A0-A15 Address Bus

A0-A15 provide a 16-bit address bus which is capable of addressing 64 Kbytes of memory and 256 I/O device ports. The I/O ports are addressed using the lower 8 address lines. A15 is the most significant address line while A0 is the least significant. During memory refresh time, the lower 7 address lines contain a valid refresh address.

D0-D7 Data Bus

D0-D7 provide an 8-bit bidirectional data bus. This bus is used for data exchanges between the CPU and memory or I/O devices. D7 is the most significant data line while D0 is the least significant.

M1/ Machine Cycle One

M1/ indicates that the current machine cycle is the one that is performing an instruction fetch. During an instruction that contains a two-byte opcode, M1/ is generated for each byte of the opcode. M1/ occurs also with IOREQ/ to indicate an Interrupt Acknowledge cycle.

MREQ/ Memory Request

MREQ/ indicates that the address bus contains a valid address for a memory read or write operation.

IOREQ/ Input/Output Request

IOREQ/ indicates that the lower 8 address lines contain a valid address for an I/O port read or write operation.

RD/ Memory Read

RD/ indicates that the Z80A wants to read data from an I/O device or from memory. The addressed device should use this signal to gate data onto the data bus.

WR/ Memory Write

WR/ indicates that the data bus contains valid data and that it may be sampled by the addressed memory or I/O device.

3.0 -- Central Processor Unit

RFSH/ Refresh

RFSH/ indicates that the lower 7 address lines contain a valid refresh address for dynamic memory systems, and the current MREQ/ signal should be used to do a refresh read to all memory. A7 is a logic zero and the upper 8 bits of the address bus contain the I register.

HALT/ Halt State* (Not implemented for Osborne 4)

HALT/ indicates that the Z80A has executed a HALT instruction and is awaiting an interrupt before execution can resume. While halted, the Z80A executes NOPs (No Operations) to maintain memory refresh operation.

WAIT/ Wait

WAIT/ indicates that the addressed I/O or memory device is not ready to transfer data. The Z80A enters wait states as long as this signal is active. The purpose of this signal is to allow memory or I/O devices of any speed to be synchronized to the speed of the Z80A.

INT/ Interrupt Request

INT/ is generated by an external device that desires to get the attention of the Z80A. An interrupt request will be honored at the end of the current instruction cycle if software has enabled interrupts and if the BUSREQ/ signal is not active. When the Z80A accepts the interrupt, an acknowledge signal (IOREQ/ during M1/ time) is sent out at the beginning of the next instruction cycle.

NMI/ Non-Maskable Interrupt* (Not implemented for Osborne 4)

NMI/ has a higher interrupt priority than INT/ and is always recognized at the end of the current instruction. It is not maskable by the software.

RESET/ Reset

RESET/ forces the Z80A program counter to zero and initializes the Z80A. The initialization consists of:

1. Disabling the interrupt enable flip-flop
2. Setting the Interrupt Vector register to 00H
3. Setting the Refresh register to 00H
4. Setting the Interrupt Mode to 0 (8080 mode)

RESET/ (cont.)

Address and data busses go to a tristate condition during reset, the control signals go inactive, and no memory refreshing occurs.

BUSRQ/ Bus Request* (Not implemented for Osborne 4)

BUSRQ/ indicates that an external device wishes to take over the Z80A address, data, and control signals.

BUSACK/ Bus Acknowledge* (Not implemented for Osborne 4)

BUSACK/ is generated by the Z80A in response to BUSRQ/ to indicate to the external requesting device that the Z80A has relinquished the bus and it may now be used by the device.

3.2 Address Drivers

The low order Z80A address lines (A0-A7) are required to drive much more circuitry than the upper address lines (A8-A15). The reason for this is that the peripheral devices (8155, 8251A, 1797), the I/O decode logic, and the display circuitry all make heavy use of these address lines.

Therefore, A0-A7 are buffered through an octal driver, 5D, located at zone B2 of sheet 1. This 74LS244 device provides a non-inverting output capable of sinking 24 milliamps. The enable pins (1 and 19), tied to ground, permanently enable each of the output drivers.

With the 74LS244 buffer in place, the Z80A is able to drive all the circuitry connected to A0-A7.

3.3 Reset Logic

In order to ensure that the Osborne 4 operates properly, a reset signal is provided which initializes all critical circuit elements to a known state. These elements include:

- o Z80A CPU
- o High Order Address Control Circuit
- o 8155 PIOs
- o 8251A USART
- o 1797 FDC
- o I/O Control Logic
- o Display Control Shift Register

3.0 -- Central Processor Unit

Only when these listed elements are properly initialized can the Osborne 4 begin to operate in a predictable, orderly fashion. The reset signal is generated in one of two ways:

1. When power is applied
2. When the reset switch (S1) is depressed

The reset logic is located at zone A3 and A4 on sheet one and consists of reset switch S1, resistors R29 and R30, diode CR3, capacitor C8, and integrated circuits 8M and 6C.

When power is first applied, C8, initially discharged, begins to charge toward +5 volts via R30. The initial voltage at C8 is zero volts; this level is then inverted twice by 8M and 6C and applied as a low signal to the RESET/ input of the Z80A, thus initializing it. RST and RST/ are also generated and fed to various logic devices as needed for initialization.

As the voltage at C8 reaches the switching threshold of 8M, the output of 8M toggles from a high level to a low level, terminating the reset condition. 8M is an 74LS14 Schmitt trigger inverter. A Schmitt trigger device is used because, unlike normal TTL devices, it has the ability to switch output states cleanly on a slowly changing input level, such as that generated by the RC input signal it receives from R30 and C8.

If switch S1 is depressed any time during the operation of the Osborne 4, C8 is discharged through R29. The machine once again enters the reset condition and stays there until the switch is released, and C8 charges to a sufficient voltage to reach the required threshold of the 8M Schmitt trigger.

The diode (CR3) provides a quick discharge path for C8 so that even if power is interrupted for a very short period of time, C8 will have sufficiently discharged to guarantee that the reset condition is forced on the CPU when power is restored.

3.4 Interrupt Logic

The Z80A may be interrupted by any of the following events:

1. A vertical sync pulse from the Display Control logic.
2. The completion of an FDC command.
3. A Service Request (SRQ/) or Attention (ATN/) signal at the IEEE 488 interface.
4. The USART ready to send or receive data.
5. An active RS-232-C Ring Indicator signal.

The described signals are all driven by open-collector devices, the outputs of which are all connected together and pulled up to +5 volts through R22, a 1K resistor at zone B3 of sheet 1. Whenever any of the above events occur, the INT/ signal to the Z80A is driven to a low level. This signals to the CPU that an external device needs servicing. If interrupts are enabled, the CPU finishes the current instruction before honoring the interrupt request. The CPU then acknowledges the interrupt request by activating IORQ/ during M1 time. This is the only instance during which these two signals ever coincide.

The Z80A can be programmed to respond to the interrupt in one of three ways:

1. Mode 0 - This mode is identical with the 8080 mode.
2. Mode 1 - CPU restarts at location 0038H.
3. Mode 2 - CPU restarts at a pointer determined in part by D0-D7 and in part by the Z80A Interrupt Vector Register.

The Osborne 4 uses Mode 2, the most powerful interrupt response mode of the three, because Mode 2 allows the software to determine the address to which the program vectors in the event of an interrupt.

Notice that the IORQ/ and M1/ signals are inputted to gate 7B at zone A3 of sheet 1. These two signals go low simultaneously only when the CPU is executing an interrupt acknowledge cycle. This results in the least significant data input to the Z80A, D0, being forced to a low state. This satisfies the condition imposed by a Mode 2 interrupt response: the least significant data bit, D0, must be a zero.

The Z80A interrupt vector register (I register), which is 8 bits wide, must be loaded with a value by the software prior to the occurrence of the interrupt. When the Z80A acknowledges an interrupt, a pattern of FEH is forced onto the data lines. This is due to the fact that D0 is forced to a zero by gate 7B and inverter 8H while the remaining data lines are pulled up to +5 volts by the resistor network RN2 at zone B3 of sheet 1. The Z80A then uses the 16 address bits formed by its own internal 8-bit I register and the external 8 data bits, D0-D7, as a pointer to vector to the interrupt service routine starting address. The pointer is used to get two adjacent bytes to form a complete 16-bit service routine starting address. Data bit D0 must be forced to a zero so that the pointer address always starts at an even location.

The following paragraphs provide the details of the various interrupts to the Z80A.

3.0 -- Central Processor Unit

3.4.1 Vertical Sync

The vertical sync pulse (VSYNC) is generated at a 60-Hz rate by the display control logic at zone C1 of sheet 6. VSYNC causes the display to start retracing back at the top of the screen after the screen has been fully scanned from top to bottom. The event must be recognized within the program flow via interrupt because that the keyboard scanning routine needs to be invoked at a regular rate, and vertical sync provides this rate. Section 10, Display Control, details the timing of this signal.

VSYNC is used as the clock input to 8L, a D flip-flop at zone C2 of sheet 5. If the flip-flop 8L is not being held in a cleared state by flip-flop 8E, the Q output of 8L will go to a one on the rising edge of VSYNC and force the INT/ line to be driven low through inverter 8B. The interrupt service routine in BIOS must do two things to service a vertical sync interrupt:

1. Verify that the interrupt was generated by VSYNC by reading the status of 8L pin 6.
2. Then reset 8L so that the next VSYNC signal out can be recognized.

In order to read the state of flip-flop 8L, the software must perform an I/O read at the appropriate port address and then examine the data read from the port. The I/O read is done by the Z80A by asserting IORQ/ and RD/ simultaneously with an address that can be anywhere in the range from 04H to 07H. Under these conditions, the FDCSEL/ signal will be generated by the one-of-eight decoder, 6D at zone A4 of page 5, and the A2/ address line will be low. These two signals are input to gate 7C at zone B3. The 7C output pin goes low and is fed along with IORD/ to gate 8C. 8C pin 10 then goes high and is sent to 8F, pin 5, enabling the output of the VSYNC flip-flop to be gated onto the data bus, line D2.

Thus, if D2 is high, it signifies to the software that there has been a vertical sync interrupt. The software must then reset the vertical sync flip-flop. It does this by performing an I/O WRITE to the same I/O address as before (04H to 07H). Gate 8C at zone B3 clocks flip-flop 8E and sets pin 9 to the state present on the D0 data line. If D0 is a zero, then 8L pin 6 is forced to a one, clearing the flip-flop 8L. If the flip-flop is left in the cleared condition, no more vertical sync interrupts can be generated because 8L can never set with its clear input low. In order to enable vertical sync interrupts, an additional I/O WRITE must be performed at the same port address, this time with D0 set to a one. The software may at any time examine the state of the vertical sync enable

flip-flop, 8E pin 8, by examining data bit D0 during an I/O READ at any I/O address from 04H to 07H. If D0 is a zero, vertical sync interrupts are disabled; if it is a one, they are enabled.

3.4.2 FDC Command Completion

The 1797 FDC has an output pin designated Interrupt Request (INTRQ) that goes high upon completion of any command sent to the FDC by the software. This signal originates at the FDC on page 9 zone B4. The signal is called FDINT on the Osborne 4 schematic. FDINT is input to gate 8F pin 2 along with an FDC interrupt enable signal from 8E pin 5 (sheet 5 zone B2). If the FDC interrupt is enabled, the output of 8F pin 3 goes low and generates an interrupt to the Z80A.

The FDC interrupt enable flip-flop is controlled by an I/O WRITE to the address range 04H to 07H with data bit D1 set to a 1. The status of the FDC enable flip-flop may be read by the software by executing an I/O READ at the same address range and examining data bit D1. If D1 is a 1, the FDC interrupts are enabled; if D1 is a 0, they are disabled.

Table 3-2 summarizes how the interrupt enable flip-flops are both written and read to by the software.

Table 3-2. Interrupt Enable Flip-Flop Control

Vertical Sync Enable F/F	Address	IOWR/	IORD/	D2	D1	D0
Command						
To Enable*	04H-07H	0	1	X	X	1
To Disable		0	1	X	X	0
Status						
Enabled		1	0	X	X	1
Disabled		1	0	X	X	0
FDC Interrupt Enable F/F						
Command						
To Enable		0	1	X	1	X
To Disable		0	1	X	0	X
Status						
Enabled		1	0	X	1	X
Disabled	04H-07H	1	0	X	0	X

* After a vertical sync interrupt is read by the software, the enable F/F must first be disabled (reset), then enabled again in order to allow the next vertical sync interrupt to be reported.

3.0 -- Central Processor Unit

3.4.3 IEEE 488 Servicing

The IEEE 488 bus is provided on the I/O board by an 8155 PIO and its associated bus drivers and receivers. The IEEE 488 bus consists of 16 active signals; 8 data and 8 bus management. Two of the bus management lines are Service Request (SRQ/) and Attention (ATN/). Whenever either of these two lines are driven low by a device using the bus, an interrupt to the Z80A occurs. ATN/ is asserted low by the device that is in charge of the bus at the moment, and, when low, signifies that commands are being sent over the data lines. SRQ/ is driven low by a device to indicate a need for service.

The ATN/ and SRQ/ signals are located at pins 11 and 10, respectively, of connector J9 (sheet 1 of the I/O board schematic). Each of the signals is directed to a separate gate B3 along with its own interrupt enable signal from the 8155 PIO. If the enable signal is at a low level, and the particular bus management signal is driven low, INT/ is asserted low at the I/O connector J3 at pin 15.

In order to service the interrupt, the Z80A must read the status of each of the signals. This is done by performing a read at any I/O address from 18H to 1FH. This causes the AUX2SEL/ signal at decoder 6D (sheet 5, zone A4) to go low. With AUX2SEL/ low and IORD/ low, gate B1 on sheet 1 of the I/O board enables octal bus driver A11, and all eight bus management lines are placed on the data bus to be read by the CPU. The software can then examine the ATN/ and SRQ/ lines and take appropriate action.

The ATN/ and SRQ/ interrupts are enabled or disabled by configuring Port C, bits PC5 and PC6, as outputs and setting them high or low appropriately. Programming the 8155 is discussed in Section 8, Input/Output board.

3.4.4 USART Servicing

The Osborne 4 is given serial I/O capability by an 8251A USART located on the I/O board. The USART is capable of operating in a full-duplex mode; it is able to transmit and receive characters simultaneously. The characters to be transmitted are parallel-loaded into the USART a byte at a time by the Z80A and transmitted by the USART serially. Received characters are serially input to the USART and transferred in parallel to the CPU. In order to minimize the processing burden on the Z80A, interrupts are generated when it is time to load the USART with a character to be transmitted, or when there is a character in the USART that has been received and needs to be unloaded to the CPU.

The USART provides two output pins, TXRDY and RXRDY, that indicate when the USART is ready to transmit a character or has a receive character ready. These signals go high when service is needed. If either signal goes high, and the corresponding interrupt enable line is low, the INT/ line will be driven low at pin 15 of the I/O board (sheet 2 of the I/O board schematic).

In order to service the USART interrupt, the Z80A must read the status register of the 8251A. This register contains a RXRDY bit and a TXRDY bit so that the software may determine the source of the interrupt and act accordingly.

The USART interrupts are enabled by the 8155 PIO on the I/O board. Bits PC3 and PC4 of port C must be configured as outputs and set high or low, as appropriate, to disable or enable the USART interrupts. The 8155 PIO programming is discussed in Section 8, Input/Output board.

3.4.5 RS-232-C Ring Indicator

In a data communications environment, the Osborne 4 can be connected to a modem. A modem converts digital data to analog signals suitable for transmission over the telephone lines. There may be times when the Osborne 4 is left unattended and is expected to answer a data call made over the phone lines from a remote data communications device that is also using a modem to transfer data. In this situation, the modem at the Osborne 4 end will provide an RS-232-C handshake line called Ring Indicator (RI). This line will be asserted to +12 volts when an external modem is being rung. The RS-232-C receiver inverts RI and presents a low level to pin 2 of gate B2 (sheet 2 of the I/O board schematic). If the RI interrupt is enabled, INT/ will be driven low through inverter A3.

In order to service the RI interrupt, the Z80A must verify that the RI signal is active. This is done by performing a read at any I/O address from 30H to 3FH. This causes the PORT3SEL/ signal at decoder 6D (sheet 5 zone A4) to be asserted low. With PORT3SEL/ and IORD/ low, gate B2 pin 6 is driven low and enables the state of RI onto data line D0. The Z80A may then read D0 and take appropriate action. If D0 is low, RI is not active; if it is high, RI is active.

The RI interrupt is enabled by the 8155 PIO. It must be configured so that bit PC2 of Port C is an output and this bit must be set low to enable the RI interrupt. 8155 programming is discussed in Section 8, Input/Output Board.

4.0 Introduction

This section describes the I/O decode and control circuitry found on sheet 5 of the schematics. The I/O decode circuit creates the select signals for the various Osborne 4 I/O ports, based on the Z80A address signals. The control logic is responsible for generating the IORD/ and IOWR/ signals in order to interface the Z80A to devices normally intended to be driven by an 8080/8085 bus system.

4.1 I/O Decode Logic

The I/O decode logic consists of a single 1 of 8 decoder, 6D, at zone A4 of sheet 5. Notice that pins 4 and 5 are the active low enable inputs and that they are driven by address lines A7 and A6, respectively. This means that A7 and A6 must both be low in order for the decoder outputs to be enabled. The decoder functions over the address range 00H to 3FH. Note also that the active-high enable input, pin 6, is driven by M1/. Since M1/ is low during an instruction op code fetch machine cycle, this signal prevents the decoder from activating any output enable lines during the time the Z80A is reading instructions. These three inputs allow I/O devices and memory to have addresses XX00H to XX3FH in common with no concern that both will be enabled onto the data bus at the same time. Table 4-1 shows the various I/O port values applicable to the Osborne 4.

4.2 Control Logic

The control logic, located at zones D2, D3, C2, and C3 of sheet 5, is designed to create special read and write signals (IORD/, IOWR/) for the I/O peripheral devices used in the Osborne 4. This ensures that the I/O timing specifications will be adequately met.

The Z80A communicates with all the I/O IC's (8155, 8251A, 1797) with IN and OUT instructions. These instructions activate the IORQ/ control line. When IORQ/ and RD/ occur together, an I/O read is performed; IORQ/ and CPUWR/ together perform an I/O write.

Section 4.0: Control/Decode Logic

The problem with the Z80A timing relationships is that the address and data hold times with respect to the trailing edge of CPUWR/ are not adequate for the I/O chips being used. This is true even though an extra wait state is added by the Z80A for all I/O read and write cycles. For this reason, additional logic is added to terminate the CPUWR/ signal early, thus giving adequate time from the trailing edge of CPUWR/ until the time that the address and data are allowed to change.

4.2.1 IORD/

IORD/ is nothing more than RD/ and IORQ/ gated together by gate 7C. When both signals are low, IORD/ goes low.

4.2.2 IOWR/

When I/O read or write cycles are not being performed, IORQ remains at a low level. IORQ is connected to the direct set inputs of D flip-flop 7E and holds the two flip-flops in a "set" condition, with the Q outputs (pins 5, 9) at a high level.

CPUWR/, connected to pin 13 of 7F, is normally high and holds the gate output, pin 11, at a high level. The pin 11 gate output is the WR/ signal. When CPUWR/ goes low, WR/ also goes low because both inputs to 7F (pins 12, 13) are low. Since this is an I/O write cycle, IORQ goes high and IORQ/ goes low at the same time that CPUWR/ goes low. This forces IOWR/ at pin 6 of 7F to go low. When IORQ goes high, it releases the direct set state of the two flip-flops. On the first rising edge of the CPU clock after CPUWR/ goes low, the first flip-flop is set, causing 7E pin 5 to go to a low level. On the next rising edge of the CPU clock, 7E pin 9 goes low and pin 8 goes high. This forces 7F pin 11 to go high, terminating the WR/ and IOWR/ pulses. Thus, the width of the WR/ and IOWR/ pulses are limited to two cycles of CPU clock, 500 nanoseconds.

Section 4.0: Control/Decode Logic

Table 4-1 I/O Port Addresses

Adr.	Port		Function	
	decoder signal	device	write	read
00-07 FLOPPY DISK				
00	FDCSEL/	FDC	Command Reg.	Status Reg.
01			Track Reg.	Track Reg.
02			Sector Reg.	Sector Reg.
03		FDC	Data Reg.	Data Reg.
04	FDCSEL/	Cmd/Stat	D2: Not Used D1: 1=En. FDC Int. 0=Dis. " " D0: 1=En. 60Hz Int 0=Dis. " "	D2: 1=60 Hz Int. Pending 0=60 Hz Int. not Pend. D1: 1=FDC Int. Enabled 0=FDC Int. Disabled D0: 1=60 Hz. Int. Enabled 0=60 Hz. Int. Disabled
05-07 NOT USED				
08-0F 8155 (SYSTEM BOARD)				
08	PIOSEL/	8155	Write to RAM	Read from RAM
09			Write to I/O reg.	Read from I/O reg.
0A-0B NOT USED				
0C			Latch RAM Address	
0D			Latch I/O Address	
			0=command/stat	
			1=port A	
			2=port B	
			3=port C	
			4=counter	
	PIOSEL/	8155	5=counter	
0E-0F NOT USED				
10-2F 8155 (I/O BOARD)				
10	AUX1SEL/	8155		Data Bus 8155 Port A Data IEEE 488 Data
11-17 NOT USED				

Section 4.0: Control/Decode Logic

Table 4-1 I/O Port Addresses (Cont.)

Adr.	Port		Function	
	decoder signal	device	write	read
18	AUX2SEL/	8155		Control/Data Bus 8155 Port B IEEE 488 Control Bus
19-1F	NOT USED			
20	PORT0SEL/	8155	Latch RAM Address	
21			Latch I/O Address	
			0=command/stat	
			1=Port A	
			2=Port B	
			3=Port C	
			4=counter	
	PORT0SEL/	8155	5=counter	
22-27	NOT USED			
28	PORT1SEL/	8155	Write to RAM	Read from RAM
29	PORT1SEL/	8155	Write to I/O Reg.	Read from I/O Reg.
2A-2F	NOT USED			
30-37	8251A (I/O BOARD)			
30	PORT2SEL/	8251A	Data Reg.	Data Reg.
31	PORT2SEL/	8251A	Command Reg.	Status Reg.
32-37	NOT USED			
38-3F	MODEM CONTROL			
38	PORT3SEL/	Modem Control		D1: 1=Data Carrier Detect 0=No Data Carrier Det. D0: 1=Ring Indicator 0=No Ring Indicator
39-3F	NOT USED			

5.0 Introduction

This section explains how the RAM and ROM sections of the Osborne 4 are organized and controlled. The explanation focuses on the ROM Enable and High Order Address Control circuitry, found on sheet 2 of the schematic, at zones D2, D3, D4, C2, C3, and C4.

The Osborne 4 has a total of 64 Kbytes of dynamic RAM, and 4Kbytes of ROM. The upper 4 Kbytes of address space, from F000H to FFFFH, are used by both the RAM and the ROM. The RAM in this area is the Screen Buffer, and contains the characters stored there by the CPU for eventual display on the CRT screen. The ROM in this area of address space contains a stored, non-volatile program that controls the basic operation of the Osborne 4.

Obviously, since both the Screen Buffer and ROM occupy the same address space, they cannot both be accessible at the same time. The following discussion explains when each is available for use by the CPU.

5.1 High Address Control Logic

When the Z80A is reset its internal program counter register is initialized at 0000H. This means the first instruction to be accessed will be read in from memory location 0000H. However, in the Osborne 4, the upper four address lines, A12-A15, are forced high when the machine is reset, so the first instruction is actually fetched from F000H.

Notice that the RST/ signal is connected to the clear input (pin 1) of the lower D flip-flop 4B, and to the preset input (pin 10) of the upper D flip-flop 4B. When RST/ is activated, pin 6 goes high and forces address lines A12-A15 high at the outputs of gates 4F. A12-A15 are then fed into the inputs of gate 4H, forcing the gate output pin 8 low. This signal becomes the chip select to the ROM, 4C, and allows the CPU to access data inside the ROM. The Z80A is then ready to fetch the first instruction, and does so by causing M1/ and RD/ to both go low, forcing gate 7B pin 10 high, which clocks 4B output pin 9 to a low state. This is the ROM Enable (ROM ENB/) signal. Gate 4A now has both of its inputs low, so its output ,(pin 6), which is ROM Output Enable (ROM OE/), goes low. At this point, gate 4A pins 9 and 10 are low, forcing output pin 8 low enabling the ROM data onto the Z80A's data bus.

Section 5.0: RAM/ROM Address Control

The first instruction to come out of the ROM must be a direct jump to some other memory location above F000H.

The Z80A program counter initializes at 0000H after reset and then increments by one address each time an instruction byte is fetched from memory. This is true even though the upper four address lines, A12 - A15, are forced high. The Z80A has no knowledge that these address lines are being forced by external circuitry, so the program counter increments as usual.

Note that after 4E pin 9 goes low, pin 8 goes high and forces input pin 2 of the lower flip-flop high via gate 4A. The next instruction fetch (M1/ and RD/ both low) clocks 4B pin 5 to a high and pin 6 to a low. This disables the condition where A12-A15 are forced high, and the next instruction fetched by the CPU will be from the actual address contained in the CPU program counter.

If by this time the program counter does not contain an address in the upper 4 Kbytes of memory (F000h and up), it will fetch an "instruction" from below F000H in RAM. However, the RAM has not been initialized or loaded yet with any meaningful data and still contains random data from power-up. Any "instruction" fetched from the RAM area will, in all probability, cause the Osborne 4 to crash. To avoid this, store a direct jump instruction at F000H, such as C3XXFX. C3 will be the first byte read after power up, followed by XX, then FX. This will cause the CPU to load the program counter with FXXXH, which is some address still within the ROM area. Then, when the next instruction is read, it will be read from FXXXH and will be a known instruction in a known location, put there by a programmer.

As long as instructions are read from F000H and above, they will come from ROM, since A12-A15 will always be high. These four address lines will cause 4B pin 9 (ROM ENB/) to stay low while M1/ and RD/ provide the clock to the 4B flip-flop. If a write to RAM locations F000H-FFFFH is attempted while executing instructions from ROM, there will be no effect. This is due to the fact that the only way to enable RAM is to force the ROM CS/ line high. This means that the current address on the address bus would have to be below F000H or ROM ENB/ would have to go high. The only way ROM ENB/ can go high is for the CPU to execute an instruction that resides below F000H.

If the CPU reads an instruction and attempts to perform a write or read operation to a RAM address below F000H, it will be allowed, since pin 8 of 4H will be high for that address, forcing the RAM CS/ low.

Section 5.0: RAM/ROM Address Control

The only way for the CPU to write to or read from the screen buffer RAM, which resides above F000H, is to execute instructions from RAM addresses below F000H. This works because M1/ and RD/ go low for the instruction fetch cycle, and because RAM instructions are being fetched and executed from below F000H. 4H, pin 8 will be held high and ROM ENB/ will be clocked continuously high, forcing RAM CS/ low. Table 5-1 summarizes the RAM/ROM selection process.

Table 5-1 RAM/ROM Selection Criteria

Instruction Location	Attempted Operation			Result
	Read	Write	Address	
P F000-FFFF	X		F000-FFFF	Read ROM
R F000-FFFF		X	F000-FFFF	No Action
O F000-FFFF	X		0000-FFFF	Read RAM
M F000-FFFF		X	0000-FFFF	Write RAM
R 0000-FFFF	X		F000-FFFF	Read Screen RAM
A 0000-FFFF		X	F000-FFFF	Write Screen RAM
M 0000-FFFF	X		0000-FFFF	Read Regular RAM
0000-FFFF		X	0000-FFFF	Write Regular RAM

5.2 ROM

The Osborne 4 uses a single 2732 4 Kbyte ROM device to provide non-volatile program storage. The Monitor ROM, 4C, is located on sheet 2 of the schematic at zone C2. The 2732 is a read-only device with a data access time of 450 nanoseconds. The ROM contains a stored program which performs the following functions:

- o Hardware Initialization at Power-up
- o Monitor Program
- o Floppy Disk Boot Program
- o Interrupt Handling for:

- 8251A Serial Port
- 1797 Floppy Disk Controller
- 8155 Parallel Ports
- 60-Hz Display Interrupt

- o Screen Handling
- o Low Level Drivers

Section 5.0: RAM/ROM Address Control

The ROM receives a chip-select when any address between F000H and FFFFH is placed on the Z80A address bus; and its contents are enabled onto the data bus when ROM CS/ is low and a memory read is being executed.

5.3 Memory Map

The Osborne 4's memory is organized into two banks, one with 64 Kbytes of RAM, and the other with 4 Kbytes of ROM. The upper 4 Kbytes of RAM are reserved for the screen buffer, where the information is kept that must be displayed on the CRT. This 4Kbyte screen buffer is only addressable by read and write instructions originating from memory locations below F000H.

When a jump is made from RAM into the area from F000H to FFFFH, instructions are then executed from ROM. This allows the ROM to "shadow" the top 4 Kbytes of RAM.

A small RAM buffer area used by the ROM is located just below F000H. This RAM is used for the program stack and for variables needed by the ROM program.

The RAM from 0000H to 0100H is a reserved area for CP/M. The remaining memory area from 0100H to the ROM buffer consists of the following CP/M components:

Transient Program Area (TPA)

Applications programs as well as CP/M utility programs are loaded into the TPA as they are used.

Console Command Processor (CCP)

This CP/M component is responsible for reading and interpreting commands from the keyboard. It is hardware independent.

Basic Disk Operating System

The disk manager and software for dynamic file allocation reside here. This is also hardware independent.

Basic Input/Output Section

This component contains basic input and output routines. This module is hardware dependent. It is altered when peripherals are added or changed.

Figure 5-1 illustrates the Osborne 4 memory map.

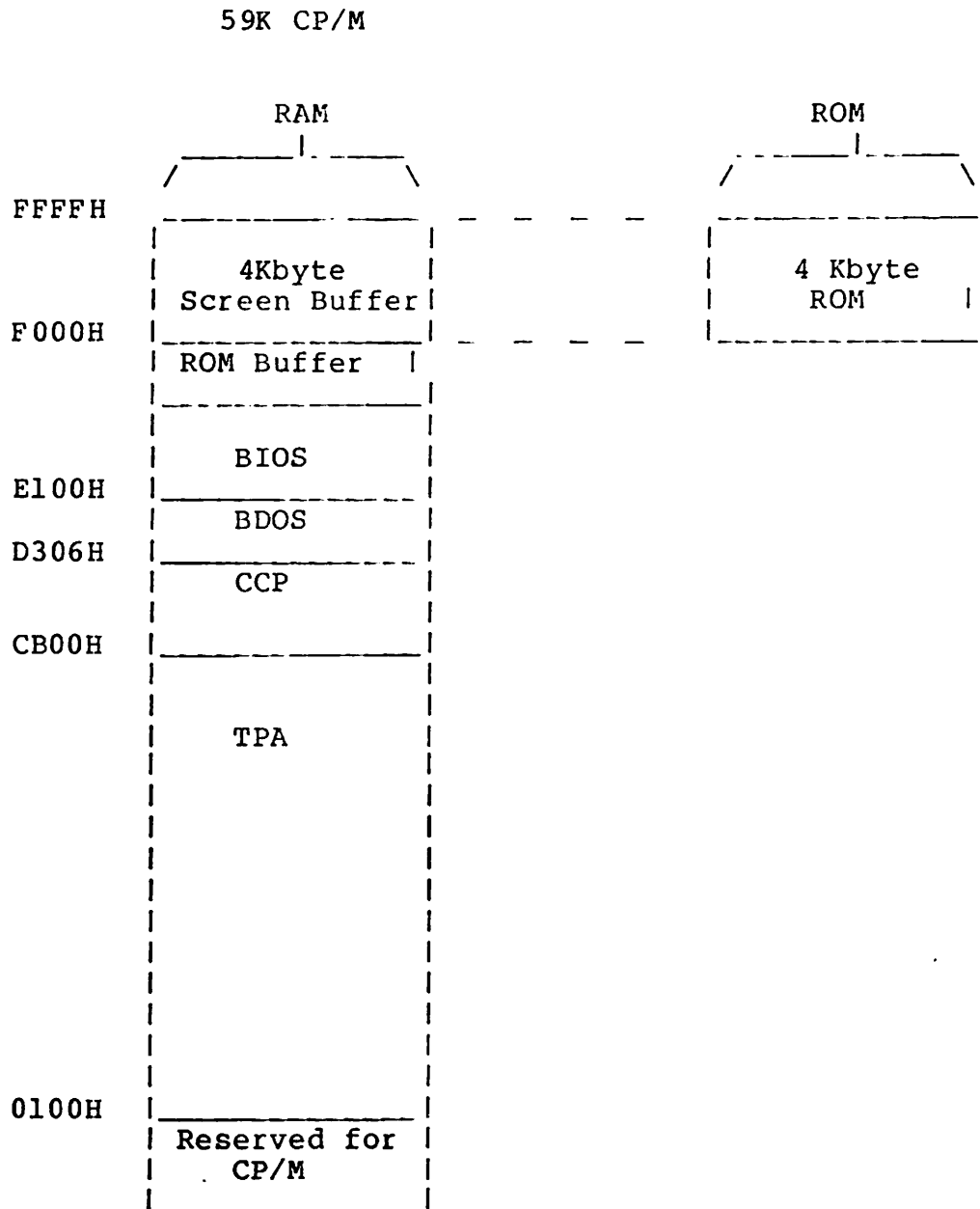


Figure 5-1 Osborne 4 Memory Map

6.0 Introduction

This section describes the function and operation of the 64 Kbyte RAM array contained within Osborne 4. The RAM is divided into two sections, as follows:

1. 60 Kbytes - for general CPU use.
2. 4 Kbytes - for screen buffer.

The RAM array consists of eight 4164-15 64K X 1 dynamic RAM integrated circuits. When power is first applied to the RAMs, each of the storage locations in each of the RAMs contains data in a random state. It is up to the program in ROM to load a meaningful program into RAM from an external device such as a floppy disk drive, or a communications port. After a program has been completely and successfully loaded in, program control may be transferred from ROM to RAM.

6.1 Dynamic RAM Control Logic

The RAM control logic generates the signals that read, write, and refresh the RAM.

When the Z80A writes to or reads from RAM, it places a 16-bit address on the address bus and activates the appropriate control signals (MREQ/, WR/, RD/). Since the RAMs have only 8 address pins, the control logic takes over and strobes the lower address bits (A0-A7) into the RAMs with Row Address Select (RAS/) signal, and the upper address bits with the Column Address Select (CAS/) signal. The address signals arrive at the RAMs 8 bits at a time through a set of 4:1 multiplexers. The RAMs have now internally latched 16 bits of address information and can begin the internal data access process for a read cycle, or can accept a write strobe for a write cycle.

The following three sections examine the memory timing for read, write, and refresh cycles. The timing requirements imposed by the 4164-15 RAMs are fairly stringent, so attention will be paid to timing margins and limits. The discussion to follow applies only to the 150-nanosecond access-time devices used in the Osborne 4. The 150 nanosecond period refers to the time required by the RAM device to output valid data from the activation of the RAS/ control signal.

Section 6.0: Dynamic Random Access Memory

6.2 Read Cycles

During a read cycle, the Z80A places a 16-bit address on the bus, then drops MREQ/ and RD/ low at the same time. After a certain amount of time, the Z80A expects to see valid RAM data at its data bus input pins. The Z80A then sets both of these control signals high, and samples the data.

Typically, 64K DRAMs need to have the row address stable as little as 0 nanoseconds before RAS/ is activated. This is easily met by the Z80A timing, since the address is guaranteed to be stable for at least 60 nanoseconds before MREQ/ goes low. MREQ/ is the signal that is used to drive the RAS/ line to the DRAMs. After RAS/ goes low, the address to the DRAMs must remain valid for at least 20 nanoseconds. This is imposed by the RAM timing specifications. Notice that the first 8 bits of address input to the DRAM are sent through the 4:1 multiplexer set. These multiplexers are controlled by the RFSH/ and COLNB lines.

During a regular read cycle, RFSH/ is high and COLNB is MREQ/ inverted. While COLNB is low and RFSH/ is high, all the "2" inputs to the multiplexers (A0-A7) appear at the outputs. When MREQ/ goes low, A0-A7 are strobed into the RAMs as the row address. However, since COLNB is the inversion of MREQ/, COLNB now goes to a high level and frees the multiplexers to select the "3" inputs to the multiplexer (A8-A15). The 20- nanosecond row address hold time is met by the delay of MREQ/ passing through a 74S04 inverter to be converted into COLNB and by the multiplexer internal propagation delay.

CAS/ must be forced low to latch the column address bits, which by now appear at the multiplexer outputs, into the RAMs. Before CAS/ can be forced low, it must be delayed by at least 30 nanoseconds from the falling edge of RAS/. This is imposed by RAM specification. This delay is produced by flip-flop 5A on sheet 2 of the schematic. When MREQ/ is high, it holds D flip-flop 5A in a reset condition, so that the Q/ output is high. This output becomes the CAS/ signal. When MREQ/ goes low, it releases the reset condition and drives the D input to a high level through inverter 3E. On the next rising edge of the CPU clock, which occurs anywhere from 20 to 85 nanoseconds later, the flip-flop is clocked and changes state, driving CAS/ low. This generates the RAS/ to CAS/ delay. There is a minimum column address hold time of 35 nanoseconds, but this is easily met, since the address out of the multiplexers will not change again until MREQ/ returns high. When CAS/ goes low, the A8-A15 address signals are latched as column address bits inside the RAMs. The RAMs now contain a complete 16-bit internal address.

Section 6.0: Dynamic Random Access Memory

Stable data appears at the DRAM data outputs a maximum of 150 nanoseconds after the RAS/ signal is activated. The Z80A requires that read data be stable at its data bus pins at least 50 nanoseconds before the falling edge of the CPU clock that occurs two clock cycles after the falling edge that activated MREQ/. Since MREQ/ goes active a maximum of 85 nanoseconds after the the first falling edge, the total data access time allowed to the RAMs by the Z80A is 500 nanoseconds (2 clock cycles) minus 85 nanoseconds (falling edge of clock to MREQ/) minus 50 nanoseconds (data setup time prior to last falling edge of clock). This results in an allowed access time of 365 nanoseconds, which allows plenty of margin for the 150 nanosecond 4164-15 RAMs used in Osborne 4.

The data bits read from the 64 Kbyte RAM array, MDATA0-MDATA7, are enabled onto the Z80A data bus by 1E, a 74LS244 octal tristate buffer. The buffer is enabled to drive the data bus when the output of gate 4A (pin 11) is low. This output is low when the inputs to the gate, RAM CS/ and MRD/ are low. MRD/ is generated by gate 6B pin 6 at zone C4 of sheet 2, and is driven low when RD is high and IORQ/ is high. RD and IORQ/ are both high during a memory read cycle. RAM CS/ is low when an instruction is not being executed out of the ROM.

6.3 Write Cycles

A write cycle is similar to a read cycle, except that the Z80A generates WR/ instead of RD/. WR/ appears considerably later in the machine cycle, so that the Z80A output data lines can stabilize before they are used by external devices. All of the address, RAS/, and CAS/ signals are generated in the same manner as they are generated during a read cycle.

The relationship of the WR/ signal to CAS/ has an important bearing on the overall memory timing. CAS/ occurs on the next rising edge of the CPU clock after MREQ/ goes low. On the very next edge of the CPU clock, which is a falling edge 125 nanoseconds later, WR/ is activated. This large amount of delay between CAS/ and WR/ forces the RAMs to enter into a read-modify-write cycle. This type of cycle is normally used in RAM error correction schemes, because valid data appears at the RAM data outputs to be corrected externally by special error-correction logic, and then routed back to the RAM data inputs in time to be written by WR/. This process is not implemented, however, in the Osborne 4. The only alternative in generating a write cycle is to generate WR/ just before CAS/. This would be an early write cycle, but is not possible in the Osborne 4 since the Z80A WR/ occurs well after the RAM control logic generates CAS/. Thus, a read-modify-write cycle is the only applicable cycle.

Section 6.0: Dynamic Random Access Memory

Even though valid data appears at the RAM outputs during this cycle, it goes nowhere, since the RAM data buffer, 1E, is not enabled during write cycles.

6.4 Refresh Cycles

Refresh is needed in order to retain data within the storage cell structure of dynamic RAM. All 128 address combinations from the address lines A0-A6 must be presented to the RAMs along with either a read, write, or RAS/ only refresh cycle every 2 milliseconds in order to meet dynamic RAM specifications.

The Osborne 4 implements this requirement in a unique manner. The screen buffer RAM is transferred into the line buffer static RAM during the time the Z80A is generating refresh address and control signals. This accomplishes two events at once:

1. The entire 64 Kbyte RAM array is refreshed.
2. A line of text is transferred from the screen buffer into the display control line buffer.

The Z80A has a built-in refresh feature. Since 128 refresh cycles must occur every 2 milliseconds, the Z80A must provide a single refresh cycle every 15 microseconds (2 ms/128). When it is time for a refresh cycle, the Z80A places a 7-bit refresh address on the address bus, then forces RFSH/ low. At this time, the 4:1 address multiplexers on sheet 3 have the RFSH/ select input low and the COLENB input low. This condition selects the "0" inputs which are then presented to the multiplexer outputs. These signals are sent as the row address inputs to the RAMs. The address consists of LTXADR0 and A6-A0. LTXADR0 is the least significant bit of the row counter for controlling the display and A6-A0 is the Z80A-supplied refresh address. Shortly after RFSH/ goes low, MREQ/ goes low. Since MREQ/ is used as the RAS/ signal into the RAMs, it strobes the address present at the multiplexer outputs into the RAMs. COLENB is the inversion of MREQ/ and the multiplexer will now select the "1" inputs and present them at its outputs. This new address becomes the column address input for the RAMs, and consists of four lines which are pulled up high and the remaining four high order row address lines for the display, LTXADR4-LTXADR1.

The next event is the generation of CAS/, generated by the D flip-flop 5A which clocks MREQ/ on the rising edge of the CPU clock. When CAS/ goes low, it strobes the address at the multiplexer outputs into the RAMs as the column address.

Section 6.0: Dynamic Random Access Memory

The end result is that every time the Z80A executes a refresh cycle, the following 16-bit address is strobed into the RAMs:

Address Bit	Signal	
15	1	\
14	1	- +5 pullups
13	1	
12	1	/
11	LTXADR4	\
10	LTXADR3	
9	LTXADR2	- Display
8	LTXADR1	Row
7	LTXADR0	Counter
6	A6	/
5	A5	\
4	A4	
3	A3	- Refresh
2	A2	Address
1	A1	
0	A0	/

The refresh address is generated from the Z80A's internal refresh register, and is incremented by one in each refresh cycle. This register counts from 00H to 7FH, for a total of 128 counts and then begins again. Thus, the Z80A controls the 7 refresh address bits, the display control logic controls the display row counter bits, and the upper 4 bits are held at a high level; the 128 bytes corresponding to a particular row displayed on the screen are being accessed from the upper 4 Kbytes of dynamic RAM memory. These bytes are available at the RAM outputs as MDATA0-MDATA7 and are then transferred into the line buffer via the display control logic. One byte is transferred each time a refresh cycle occurs.

Refer to Section 10 for a more detailed explanation of how the data is transferred from the dynamic RAM to the display line buffer.

7.0 Introduction

This section explains how the keyboard is controlled. The Osborne 4 scans the keyboard for a key depression 60 times per second, whether a key is depressed or not. It does this through a scanning routine in software that is triggered every time a 60 Hz interrupt occurs. The display control logic generates a vertical sync pulse 60 times a second. This pulse is used to interrupt the Z80A and initiate keyboard scanning.

The keyboard is organized into an 8 by 8 matrix, or 64 keys total. The scanning of the keyboard is done by sending out a code on the 8 lines corresponding to the 8 rows of the matrix and reading the resultant code returned on the 8 lines representing the 8 columns of the matrix. The software then determines from the column code exactly which key was depressed.

After a key is depressed, the software must go through a debounce routine to make sure that the key closure (or release) is really valid. The transitions that occur at the beginning of a key closure or at the end of a key release must be eliminated.

A character entered from the keyboard must eventually be deposited in the screen buffer so that it can be displayed on the CRT. But before it does, two basic things must be determined:

1. The key must be decoded into the proper character.
2. The proper video attributes must be given to the character.

7.0.1 Key Decoding

When a key is depressed, any of the following keyboard control options may be in effect:

- o Lower Case
- o Shift
- o Control
- o Control/Shift
- o Alpha Lock

For this purpose, there are five separate decode tables in ROM, one for each of the control options listed above. The software must determine which of the five options is in effect at the moment the key is depressed, then go to the proper decode table. The decode table translates the column code into the proper character code. It is this

Section 7.0: Keyboard Control

character code that eventually makes it to the character generator ROM in the display control section for translation into an actual character on the screen.

7.0.2 Video Attributes

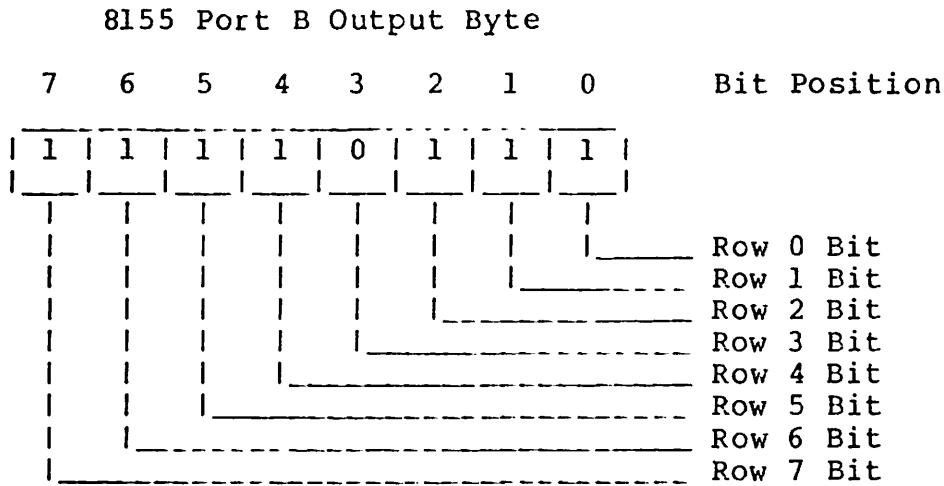
Once the proper character code has been determined for the key depression, the video attributes of the character must be assigned. The attribute list for a character is as follows:

- o Reverse/Normal Video
- o Alternate/Normal Character Set

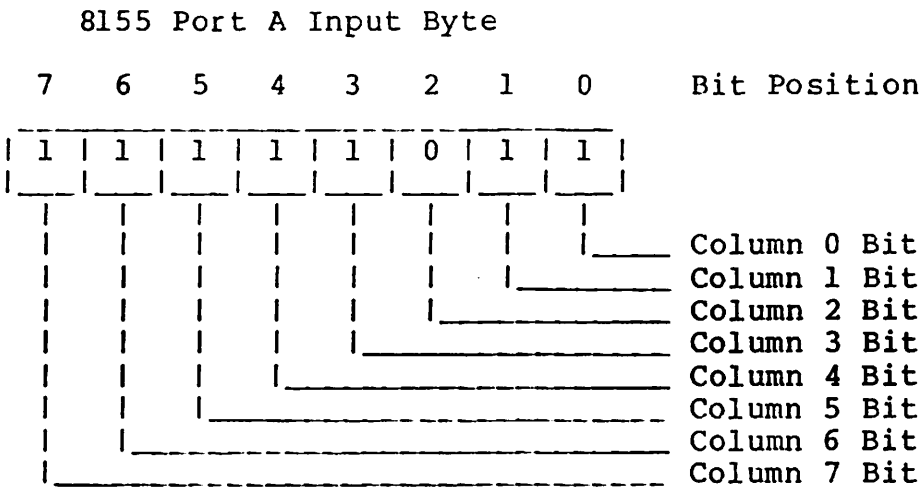
In order to make sure the character gets tagged with the proper attribute, there are five 64-byte tables maintained in ROM, one for each of the five attributes listed above. Pointers to these tables are maintained in RAM, so that the proper type of attribute is given to a particular character as it is entered from the keyboard. The software then takes care of entering the proper character into the screen buffer and controlling the ALT CHAR SET/ (Alternate Character Set) and 256 CHARS/ (256 Character Set) control lines to the display control logic. These two signals are responsible for determining if a standard or alternate character set gets displayed, and whether reverse video is to be displayed on an entire screen basis or not.

The central element in controlling the keyboard is the 8155 PIO. Through software control, it generates the row address codes to the keyboard and reads the resulting column address codes. Port B of the 8155 is configured as an 8 bit output port and drives the 8 row address lines through open-collector buffers 1M and 1N at zone D2 of sheet 4. The software sets low one of the 8 bits of the row address, with all the rest of the bits set high, then executes an OUT instruction. This low bit is then read by Port A of the 8155 with an IN instruction. The location of the low bit corresponds to a particular key depression location along that row of the keyboard. The software then sets low another bit of the row address and looks at the returning column address to see if a key depression has been made on the next row. In this manner, all 64 keys of the keyboard are scanned.

Since all 8 column address lines at Port A of the 8155 are pulled up to +5 volts, the bits will all be high if there is no key being depressed. A low bit in the column address indicates a key depression. A high bit signifies no key depression. More than one bit may be set low in the row address. Figure 7-1 shows the generation of a row address and the resulting column address:



The Row 3 bit has been set low in order to test row 3 for a key depression.



The column 2 bit is low, indicating that the row 3, column 2 key has been depressed. This is the "R" key on the Osborne 4 keyboard.

Figure 7-1 Row and Column Keyboard Addresses

Section 7.0: Keyboard Control

7.1 8155 PIO

The 8155 on the system board controls the row and column addressing for the keyboard and controls the video attributes of the characters displayed on the CRT. This same 8155 provides diskette density control and drive select for the disk drives, and a beep enable signal for driving the speaker. The following information shows how the 8155 is controlled by the software to perform these functions.

7.1.1 8155 Overview

The 8155 has the following features:

- o 256 Bytes of Static RAM
- o Three I/O Ports

- Port A - 8 bits
 - Port B - 8 bits
 - Port C - 6 bits

- o 14 Bit Timer/Counter

The 8155 also has the following internal registers:

- o Command/Status
- o General Purpose I/O Port A
- o General Purpose I/O Port B
- o General Purpose I/O Port C
- o Timer Count Low Order Bits
- o Timer Count High Order Bits and Timer Mode

7.1.2 Command Register

In order to use the 8155, the system software must program it by writing to the command register. This register defines the nature of Ports A, B, and C as to whether they are to be used as input or output ports, and also controls the timer operation. The I/O Port registers and Timer registers may then be written to or read.

7.1.3 I/O Addressing

The 8155 is designed to function specifically with the 8085 microprocessor. The 8085 multiplexes address and data on its lower 8 address lines. During the early part of the machine cycle, address is present along with an Address Latch Enable (ALE). An external latch is then needed to latch the address, because after ALE occurs, the same 8 lines are used for the data bus. In this case, the 8155 latches the address on the bus in an internal address

latch with ALE and then uses the Z80A WR/ or RD/ control signals to accept data from or provide data to the processor.

In the case of Osborne 4, the 8155 interfaces to a Z80A. Since the Z80A does not have this ALE signal, a two-step operation is performed. The 8155 is regarded as having two separate I/O addresses. One is used for latching address information that is presented on the data lines (D0-D7), and the other is used for reading or writing the 8155 register whose address was just latched on the same data lines. Table 7-1 illustrates the I/O addressing of the 8155.

Z80A I/O Address	WR/	RD/	Function
0DH	X		Latch D0-D7 as 8155 Register Address
			Data Bus Bits 8155 Register Addressed 7 6 5 4 3 2 1 0 X X X X X X 0 0 0 X X X X X X 0 0 1 X X X X X X 0 1 0 X X X X X X 0 1 1 X X X X X X 1 0 0 X X X X X X 1 0 1 _ _ _ _ _ _ _ Command/Status I/O Port A I/O Port B I/O Port C Low Timer Bits High Timer Bits
09H	X		Write to Addressed 8155 Register
09H		X	Read From Addressed 8155 Register
0CH	X		Latch D0-D7 as 8155 RAM Address
08H	X		Write to Addressed RAM
08H		X	Read From Addressed RAM

Table 7-1. System Board 8155 Addressing

Section 7.0: Keyboard Control

7.1.4 Row Address

Port B of the 8155 is configured as an output port and, under control of software, sends an 8-bit row address to the keyboard. These lines are driven out non-inverting open-collector buffers 1M and 1N.

7.1.5 Column Address

Port A of the 8155 is configured as an input port and under control of the software, reads the keyboard 8-bit column address. All eight bits are normally pulled up to +5 volts by a set of 10K resistors RN1. The only time a bit is low is when one of the row address lines goes low and a key is being depressed. The key depression completes the circuit and allows the row bit to be presented at Port A.

7.1.6 Control Signals

The 8155 generates several control signals described in the following list:

BEEP ENB/ Beep Enable.

This signal enables a 1.56 KHz tone generated by the display control circuitry. Gate 6K at zone A4 of sheet 8 allows this tone to activate the audio alarm transducer when BEEP ENB/ is low.

256 CHARS/ 256 Character Set.

This signal is used as the select control signal for the character select multiplexer, 4M at zone D3 of sheet 8. When this signal is active and the character generator ROM is a 2732, a full 256 character set may be displayed.

ALT CHARS/ Alternate Character Set.

This signal selects between two 128 character sets stored in the character generator ROM when the ROM being used is a 2732.

FSERVO Floppy Disk Servo.

This signal is reserved for future applications.

Section 7.0: Keyboard Control

DDEN/ Double Density Enable.

This signal is used by the 1797 floppy disk controller chip and by the 9216B data separator circuits when reading and recording disks in double density format.

DSEL1/ Drive Select One.

This signal selects floppy disk drive number one.

DSEL2/ Drive Select Two.

This signal selects floppy disk drive number two.

8.0 Introduction

The I/O board provides the Osborne 4 with an expansion capability. This board plugs into the main processor board and supplies the following functions:

1. IEEE-488/Centronics Parallel Interface
2. Synchronous/Asynchronous Serial Interface

8.1 Parallel Interface

The parallel interface provided on the I/O board actually consists of two separate 8 bit I/O ports. One port provides a bidirectional data bus and the other provides a bidirectional control bus. These ports are general-purpose and may be programmed to be either IEEE-488 or Centronics compatible. At the heart of the parallel interface is an 8155 PIO, which uses ports A and B for the data and control busses.

8.1.1 8155 PIO

The principal use of the 8155 on the system board is to provide for the IEEE-488 and Centronics interfaces. The IEEE-488 interface is the standard used to interface programmable and non-programmable instruments, computers, and peripherals necessary to build an instrumentation system. The Centronics interface is the industry-wide standard parallel printer interface. Besides providing for these interfaces, the 8155 also generates several control signals.

8.1.2 8155 Overview

The 8155 has the following features:

- o 256 Bytes of Static RAM
- o Three I/O Ports
 - Port A - 8 bits
 - Port B - 8 bits
 - Port C - 6 bits
- o 14 Bit Timer/Counter

Section 8.0: Input/Output Board

The 8155 also contains the following internal registers:

- o Command/Status
- o General Purpose I/O Port A
- o General Purpose I/O Port B
- o General Purpose I/O Port C
- o Timer Count Low Order Bits
- o Timer Count High Order Bits and Timer Mode

8.1.2.1 Command Register

In order to use the 8155, the system software must program it by writing to the command register. This register defines the nature of Ports A, B, and C as to whether they are to be used as input or output ports, and also controls the timer operation. The I/O Port registers and Timer registers may then be written to or read appropriately.

8.1.2.2 I/O Addressing

The 8155 is designed to function specifically with the 8085 microprocessor. The 8085 multiplexes address and data on its lower 8 address lines. During the early part of the machine cycle, the address is present along with an Address Latch Enable (ALE) signal. An external latch is then needed to latch the address, because after ALE occurs, the same 8 lines are used as the data bus. What the 8155 does in this case is to latch the address on the bus in an internal address latch with ALE and then use the Z80A WR/ or RD/ control signals to accept data from or provide data to the processor.

In the case of Osborne 4, the 8155 interfaces to a Z80A. Since the Z80A does not have this ALE signal, a two-step operation is performed. The 8155 is regarded as having two separate I/O addresses. One is used for latching address information that is presented on the data lines (D0-D7) , and the other is used for reading or writing the 8155 register whose address was just latched on the same data lines. Table 8-1 illustrates the I/O addressing of the 8155..

Section 8.0: Input/Output Board

Table 8-1. I/O Board 8155 Addressing

Z80A I/O Address	WR/	RD/	Function
21H	X		Latch D0-D7 as 8155 Register Address
			Data Bus Bits 8155 Register Addressed 7 6 5 4 3 2 1 0 X X X X X 0 0 0 X X X X X 0 0 1 X X X X X 0 1 0 X X X X X 0 1 1 X X X X X 1 0 0 X X X X X 1 0 1
			Command/Status I/O Port A I/O Port B I/O Port C Low Timer Bits High Timer Bits
29H	X		Write to Addressed 8155 Register
29H		X	Read From Addressed 8155 Register
20H	X		Latch D0-D7 as 8155 RAM Address
28H	X		Write to Addressed RAM
28H		X	Read From Addressed RAM

8.1.2.3 Port A

Port A of the 8155 is used as an output data bus. It can be seen by referring to the schematic that Port A drives the parallel data interface through a set of open-collector drivers that are terminated in 3.3K ohm pullup resistors and 6.8 K ohm pulldown resistors. I/O data lines DI01-DI08 are driven either by port A or by the external device (Centronics compatible printer or IEEE-488 device). The software must control who drives the bus when. The way that the data makes it back to the Z80A is through the octal 74LS244 bus driver A7.

Activation of the IORD/ and AUX1SEL/ lines turn on the buffer and allow it to drive the Z80A data bus input. AUXSEL1/ is driven low when the Z80A address is 10H.

To write a pattern of 55H to the IEEE-488 data lines, you would first write a data pattern of 01H to the I/O address 21H. This latches an I/O address of 01H internal to the

Section 8.0: Input/Output Board

8155, which is the address for port A. Then you would perform an I/O write to address 29H with a data pattern of 55H. This puts 55H on the port A outputs. If you wanted to then read that pattern, you would perform an I/O read to location 10H, which turns on the A7 buffer and allows the data onto the Z80 data bus.

8.1.2.4 Port B

Port B works exactly like port A, except the I/O addressing is different. Also, this port is used as a control port instead of a data port. An example will be given to illustrate the I/O addressing differences.

If, for example, you wanted to write a pattern of 55H to the IEEE-488 control lines, you would first write a data pattern of 02H to the I/O address 21H. This latches an I/O address of 02H internal to the 8155, which is the address for port B. Then you would do an I/O write to address 29H with a data pattern of 55H. This puts 55H on the port B outputs. If you wanted to then read that pattern, you would do an I/O read to location 18H, which turns on the A11 buffer and allows the data onto the Z80A data bus.

8.1.2.5 Port C

Port C is used as a control signal port. Data is written to port C as follows. First, a data pattern of 03H is written to I/O address 21H. This latches an I/O address of 03H internal to the 8155, which is the port C address. The desired data pattern to control the proper control line is then written at I/O address 29H. The data pattern is then placed on port C control lines. The control signals generated by port C are defined as follows:

ENB SRQ INT/ Enable Service Request Interrupt.

This signal is directed to gate B3, pin 6 along with Service Request (SRQ/). When both signals are low, the Z80A receives an interrupt.

ENB ATN/ INT/ Enable Attention Interrupt.

This signal is directed to gate B3, pin 8, along with Attention (ATN/). When both signals are low, the Z80A receives an interrupt.

Section 8.0: Input/Output Board

ENB XMT INT/ Enable Transmit Interrupt.

This signal is directed to gate B3 pin 11 along with TXRDY/ from the USART. When both are low, an interrupt is sent to the Z80A.

ENB RCV INT/ Enable Receive Interrupt.

This signal is directed to gate B3 pin 3 along with RXRDY/ from the USART. When both are low, an interrupt is sent to the Z80A.

ENB RING INT/ Enable Ring Interrupt.

This signal is directed to gate B4 pin 11 along with the RS232C Ring Indicator signal. The Z80A is interrupted when Ring Indicator is active.

SEL INT CLK Select Internal Clock.

This signal controls multiplexer A5 (found on sheet 2 of the I/O schematics, zone B3) and, when high, selects the clock generated by the 8155 as the USART baud rate clock. This is considered to be an internal source of clock.

8.1.2.6 Timer

The 8155 has a built-in timer which may be configured by the software. In the case of the I/O board, this timer is used to provide a baud rate clock to the USART, which also resides on the I/O board.

First, the timer mode and count (divisor value) length must be written into the two timer registers. An I/O address of 21H must be generated along with a pattern of 04H on the data bus from the Z80A. This latches an I/O address internal to the 8155 of 04H, which is the first timer register. Then an I/O address of 29H is generated with an appropriate data pattern to write the desired timer mode and most significant bits of count length. Next, writing to 21H with data 05H latches the address of the second timer register within the 8155. Writing to 29H may then be done with data to write the desired least significant bits of count length. Consult the 8155 data sheets for specific details on the bit patterns for mode and count length.

Section 8.0: Input/Output Board

The 8155 generates a clock at pin 6 which has a period as specified by the count-length values written into the two timer registers. This clock is used by the 8251A USART and may be 1X, 16X, or 64X the baud rate at which the USART is to operate. The timer output is derived from the 4 MHz CPU clock.

8.1.3 IEEE-488 Interface

The following is a brief description of each of the signals supported by an IEEE-488 interface on the Osborne 4:

DI08-DI01 - (Data).

These are the bidirectional data lines for the IEEE 488 data.

ATN/ - (Attention).

The controller in charge asserts this line. When low, it signifies that commands are being sent over the data lines. When ATN is high, data is being carried over the data lines.

SRQ/ - (Service Request).

This is set low by a device to indicate a need for service.

IFC/ - (Interface Clear).

This signal is sent by the system controller to reset the system to a known state. The system then becomes the controller in charge.

NDAC/ - (Not Data Accepted).

The device receiving data sets this line high when it has latched data from the I/O lines.

NRFD/ - (Not Ready for Data).

This signal is sent by the device receiving data to indicate readiness for accepting the next byte.

DAV/ - (Data Valid).

This line is controlled by the source device to indicate that valid data is present on the data bus.

REN/ - (Remote Enable).

This line is sent by the System Controller to select control either from the console or from the IEEE bus.

EOI/ - (End or Identify).

If ATN is high, , this signal indicates the end of a message block. If ATN is true, the controller is requesting a parallel poll.

8.1.4 Centronics Interface

The following is a brief summary of the signals used in the standard Centronics parallel printer interface:

D08-D01 - (Data).

These lines provide data to the printer from Osborne 4. All data patterns with a high level in bits 6 or 7 are considered to be printable characters. All data patterns with a low level in both bits 6 and 7 are regarded as special control characters.

DATA STB/ - (Data Strobe).

This signal is used to clock the data from Osborne 4 into the printer buffer. The pulse width is a nominal 1 microsecond.

PRIME/ - (Prime).

This signal, when low, initializes the printer logic and clears the printer buffer. It is generated whenever the Osborne 4 is reset.

BUSY - (Busy).

This signal, when high, indicates that the printer cannot receive data. It may indicate a buffer full, paper empty, or fault condition.

ACK/ - (Acknowledge).

This active low pulse indicates that the printer has accepted a print character or command.

Section 8.0: Input/Output Board

SEL/ - (Select).

This signal, when low, indicates that the printer is selected, or on-line, and may be used for printing.

FAULT/- (Fault).

This active-low signal indicates that the printer has detected an error condition, such as paper empty, deselect, etc.

8.2 Serial Interface

The serial interface provided on the Osborne 4 is controlled by an 8251A USART. The 8251A may be programmed by software to operate in full-duplex mode using any serial data transmission protocol desired. The USART may be used in an asynchronous mode, using an internally generated baud rate clock, or it may be operated synchronously, with the clock being provided by an external source. If an internal clock is chosen, both the transmitter and receiver must operate at the same baud rate; when external clocks are used, the receive clock is allowed to run at a different rate than the transmit clock.

The USART accepts data characters from the Z80A in parallel format and converts them into a continuous serial data stream for transmission. Simultaneously, it can receive a serial data stream and convert it into parallel characters for the Z80A. The USART will signal the Z80A whenever it has received a new data character or whenever it needs a new character to transmit. The Z80A can read the complete status of the USART at any time.

8.2.1 USART Initialization

Prior to starting data transmission or reception, the 8251A must be loaded with a set of control words generated by the Z80A. These control words define the complete operation of the USART and must immediately follow a reset operation. There are two control words, both of which are written at I/O address 31H:

1. Mode Instruction
2. Command Instruction

The format of the mode instruction defines the general operating characteristics of the USART. The first control word after reset operation is interpreted by the USART as a mode instruction. Once the mode instruction has been

written to the USART, subsequent writes to 31H will be the command instruction. The command instruction controls the actual operation of the USART and may be written any time during the operation of the device. To be able to rewrite the mode instruction, a reset must occur first or an 8251A internal reset must occur. Consult the 8251A data sheets for the format of the mode and command instructions.

8.2.2 USART Servicing

Once the USART has been programmed as desired, the Z80A must be able to read the internal status of the chip. This is done by performing an I/O read at 31H.

When data characters are received and assembled by the USART, the RXRDY output at pin 14 goes high and will generate an interrupt to the Z80A if the ENB RCV INT/ line is low. The Z80A responds to this interrupt by reading the contents of the USART receive holding register, located at I/O address 30H. In a similar manner, when the USART has completely shifted out a character previously written to it by the CPU, the TXRDY output at pin 15 goes high and will cause an interrupt if the ENB TX INT/ line is low. The Z80A responds by writing the next character to be transmitted into the USART transmit holding register, located at I/O address 30H.

8.2.3 RS232C Interface

The 8251A communicates to the outside world through the industry standard RS-232-C interface. Following is a brief summary of the signals supported.

TXD Transmit Data.

This output provides a path for serial transmit data from the Osborne 4 to an external device.

RXD Receive Data.

This input provides a path for serial receive data from an external device to the Osborne 4.

DTR Data Terminal Ready.

This signal indicates the state of the Osborne 4. When active, it signifies that the Osborne 4 is ready to accept data from an external device.

Section 8.0: Input/Output Board

DSR Data Set Ready.

This signal, when active, indicates to the Osborne 4 that the external device with which it is communicating is operational.

CTS Clear to Send.

This signal from the external device indicates that the Osborne 4 may now transmit data.

RTS Request to Send.

This signal indicates to the external device that the Osborne 4 desires to transmit data to it.

DCD Data Carrier Detect.

This signal, normally from a modem (or DCE), indicates that the modem is receiving a carrier of sufficient quality for proper data reception.

RI Ring Indicator.

This signal, normally from a modem (or DCE), indicates that a ringing condition has been detected and that the Osborne 4 should answer by activating the DTR line.

TXC Transmit Clock.

This clock is provided by an external device for synchronous data transmission capability. It is connected to the USART's TXC input if the Select Internal Clock signal from the 8155 is at a low level.

RXC Receive Clock.

This clock is provided by an external device for synchronous data transmission. It is connected to the USART's RXC input if the Select Internal Clock signal from the 8155 is at a low level.

Section 9.0: Floppy Disk Controller

9.0 Introduction

The Osborne 4 comes equipped with two 5-1/4 inch, half-height, ANSI-standard disk drives. These drives accept standard 5-1/4 inch, double-sided, double-density, soft-sectored diskettes. The diskettes may be formatted in either single-density (FM) or double-density (MFM) format.

The main processor board contains a 1797 Floppy Disk Controller (FDC) chip, referred to as the FDC. The FDC permits disks to be read and written in standard IBM 3740 (FM) or IBM System 34 (MFM) format with 128, 256, 512, or 1024 bytes per disk sector. Variations in the IBM format are also possible to a limited degree if certain sector size and gap size limitations are met. The result is a flexible, adaptable system which supports the following industry formats:

Osborne 1	Single/Double Density
IBM PC	Double Density (CP/M-86)
DEC VT180	Double Density
Xerox 821	Single Density
Cromemco Mini-Disk	Single Density

The formatted disk capacities supported by the Osborne 4 are outlined in Table 9-1.

Table 9-1. Diskette Capacities

	OCC Single Density	OCC Double Density	Xerox Single Density	IBM Double Density	DEC Double Density
Bytes/Diskette	100K	200K	90K	160K	180K
Bytes CPM Space	92K	185K	83K	156K	171K
Bytes/Sector	256	1024	128	512	512
Sectors/Track	10	5	18	8	9
Tracks/Diskette	40	40	40	40	40
SectorTranslate	2 to 1	2 to 1	5 to 1	1 to 1	2 to 1

The Osborne 4 Floppy Disk Drive electronics can be discussed under the following headings:

1. Z80A Interface
2. 1797 FDC Operation
3. Floppy Disk Interface
4. Write Precompensation
5. Data Separation

Section 9.0: Floppy Disk Controller

9.1 Z80A Interface

The Z80A interface to the floppy disk drive electronics is done via the 1797 FDC. The FDC provides a bidirectional 8-bit data bus, 2 address-line inputs, chip-select input, read- and write-control inputs, and an interrupt-request output for use by the Z80A.

The FDC receives a chip select signal whenever the Z80A issues an IORD/ or IOWR/ signal with an address in the range from 00H to 03H. When the Z80A issues one of these addresses, it uses the eight data lines D0-D7 to transfer data, status, and control words into, or out of, the FDC. The registers inside the FDC are selected according to Table 9-2.

Table 9-2. FDC Internal Register Addressing

FDC SEL/	A2	A1	A0	IORD/	IOWR/
0	0	0	0	Status Reg	Command Reg
0	0	0	1	Track Reg	Track Reg
0	0	1	0	Sector Reg	Sector Reg
0	0	1	1	Data Reg	Data Reg

Status Register

The eight-bit status register holds device status information. The meaning of each of the bits depends on which type of FDC command was previously executed.

Command Register

This register holds the command that is presently being executed.

Track Register

This register holds the track number of the current Read/Write head position.

Sector Register

This register holds the address of the desired sector location.

Section 9.0: Floppy Disk Controller

Data Register

This register is used as a holding register during disk read/write operations.

9.2 1797 FDC Operation

The following paragraphs briefly summarize the 1797 FDC operation. For detailed information, consult the 1797 IC Specifications. data sheets, included in Appendix C of this document.

9.2.1 FDC Commands

The FDC will accept eleven different commands from the Z80A. These commands are as listed below and are of four types:

Type I - Restore, Seek, Step, Step-In, Step-Out

Type II - Read Sector, Write Sector

Type III - Read Address, Read Track, Write Track

Type IV - Force Interrupt

9.2.1.1 Type I Commands

Restore

This command causes the drive to seek Track 00.

Seek

This command causes the drive to seek until the head is positioned at the track specified.

Step

This command causes the head to step by one track in the same direction as in the previous step command.

Step-In

This command causes the head to step towards the inside of the diskette in the direction of the higher track numbers.

Step-Out

This command causes the head to step towards the outside of the diskette in the direction of the lower track numbers.

Section 9.0: Floppy Disk Controller

9.2.1.2 Type II Commands

Read Sector

This command cause the drive head to locate the specified sector. The data within the sector is then transferred to the FDC. The FDC notifies the Z80A that a byte is ready by activating the Interrupt Request line and setting the Data Request bit in the status register.

Write Sector

This command locates the specified sector, and then transfers data from the FDC to the diskette. The Z80A is notified to send a new byte to the FDC by activating Interrupt Request and setting the Data Request in the status register.

9.2.1.3 Type III Commands

Read Address

This command causes the next encountered ID field to be read in from the diskette. The track address, side number, sector address, sector length, and two CRC bytes are included in this field. The Z80A is notified once again through the IRQ/ and DRQ/ functions.

Read Track

This command reads in the entire track, starting with the first index pulse encountered and stopping after the next index pulse.

Write Track

This command writes an entire track, starting with the first index pulse, and ending with the next.

9.2.1.4 Type IV Commands

Force Interrupt

This command allows the Z80A to terminate any command in progress.

Section 9.0: Floppy Disk Controller

9.2.2 FDC Status

The status register permits the Z80A to check the progress of FDC operations. The status bits have distinct meanings for different commands. When a command is initiated, the Busy Bit is set and the other status bits are cleared or updated.

9.2.3 FDC Read Operation

For read operations, the FDC requires two input signals. The first is RAW READ/, at pin 27, which is a 250 nanosecond pulse that occurs every time there is a flux transition. The second is RCLK, at pin 26, which operates at 250 KHz when in double density mode, or 125 KHz when in single density mode. Both of these signals are provided by the data separator, discussed in a later section.

9.2.4 FDC Write Operation

For write operations, the FDC provides four main output signals. WRITE GATE, at pin 30, allows write current to flow into the Read/Write head. The WRITE DATA signal, at pin 31, along with LATE and EARLY at pins 18 and 17 are output to a write precompensation circuit, which is covered in a later section. The write precompensator positions the data correctly and then outputs it to the drive.

9.3 Floppy Disk Drive Interface

This section describes the signals which flow between the Osborne 4 floppy disk drive electronics and the disk drive itself. These signals are the industry-standard ANSI interface common to all standard 5-1/4 inch disk drives.

INDEX/ Pin 8

This signal is provided by the drive each time an index hole is sensed at the index photo detector. Normally, this is an active-low signal; it occurs once per diskette revolution (every 200 milliseconds).

DRIVE SEL 1/ Pin 10

When this signal is active, Drive number one (A) is selected for read, write, and control operations.

Section 9.0: Floppy Disk Controller

DRIVE SEL 2/ Pin 12

When this signal is active, Drive number two (B) is selected for read, write, and control operations.

DIRECTION/ Pin 18

This line defines the direction of motion the read/write heads will take when the STEP/ line is pulsed. A high signal defines the direction as "out" toward lower track numbers, while a low level defines the direction to be "in" or toward higher track numbers.

STEP/ Pin 20

This signal causes the heads to move in the direction specified by the DIRECTION/ line. This signal is active low.

WRITE DATA/ Pin 22

This line provides the data to be written onto the diskette. Each transition from a logic one to a logic zero level will cause current in the read/write heads to be reversed thereby writing a data bit. This line is enabled by WRITE GATE/ being active. WRITE GATE/ must be inactive during a read operation.

WRITE GATE/ Pin 24

This active low signal enables data to be written onto the diskette. When the signal is inactive, the read data logic on the drive is enabled.

TRACK00/ Pin 26

When this signal is low, it indicates that the drive's read/write heads are positioned at track 00 (the outermost track).

WRITE PROTECT/ Pin 28

This signal, when low, indicates that the diskette being used is write protected. This condition prevents the FDC from writing data onto the diskette.

Section 9.0: Floppy Disk Controller

READ DATA Pin 30

This line provides "raw data" (data and clock pulses together) as detected by the drive electronics. Normally, this signal is at a high level; it goes low to indicate the presence of data.

SIDE SELECT/ Pin 32

This signal defines which side of a two-sided diskette is being used. A high level selects side 0. When switching from side to side, a delay is needed before a read or write operation can be performed.

NOTE: All odd-numbered pins on the connector are grounded.

9.4 Write Precompensation

Write precompensation improves the data-handling reliability margin in the disk system. As different combinations of bit patterns are written to the disk, the bits have a tendency to shift away from a nominal position in the data stream. This shift can be either late or early with respect to the nominal position. Precompensation is accomplished by writing a bit in the opposite direction to its known shift.

During write operations to the disk, the FDC provides two signals to assist in write precompensation. These signals are EARLY and LATE, at pins 17 and 18 of the FDC. EARLY is active high when the WD pulse at pin 30 is to be written early. LATE is high when the WD pulse is to be written late. The signals are both low when the WD pulse is to be written at its nominal position.

The 74LS166 shift register, 7L, at zone B1 of sheet 9 provides the time shifting of the write data in accordance with the EARLY and LATE control signals. When there is no data being written to the diskette, the WD signal at pin 31 of the FDC is low. This is inverted and presented to the parallel load control input (pin 15) of 7L as a high level. This allows the register to serially shift out the level present at the serial data input, pin 1. Since pin 1 is always low, the data which is shifted through the register and which finally appears at pin 13, the QH output, is a constant low level. When the WD line goes high, it takes the parallel-load control input low and loads the data appearing at the parallel inputs. If EARLY and LATE are both low, the F parallel input is high, and the QF output goes high on the next rising edge of clock. The QH output goes high two clocks cycles later. This is normal nominal write precompensation.

Section 9.0: Floppy Disk Controller

If EARLY is high and LATE is low, then the QG output goes high on the WD data transition and QH goes high one clock cycle later. This is early write precompensation. If LATE is high and EARLY is low, the QE output goes high with WD and the QH output goes high three clocks later. This is late write precompensation.

The time positioning of the write data can be shifted by 250 nanoseconds early or late from nominal. For this to be possible, the data has to experience some absolute delay. The result is that nominal data write precompensation is delayed by two clock cycles (500 nanoseconds) from the time of its actual occurrence at the output pin of the FDC.

9.5 Data Separation

The READ DATA input from the drive contains both data and clock information. The data separator circuitry extracts one from the other. This is especially critical in reading diskettes which have been recorded in double density format, due to the occurrence of peak-shift (time distortion during playback from magnetic recordings) per bit cell. This distortion is partially overcome by write precompensation.

The data separator circuitry centers around the 9216B data separator chip, 6L. This chip eliminates many components used in previous designs. A reference clock of 4 MHz is provided on pin 3, and is divided inside the 9216B to provide an internal clock. The internal clock runs at 4 MHz for double density operation and at 2 MHz for single density operation.

The 9216B detects the leading edge of the READ DATA pulses coming into it on pin 1 and adjusts the phase of its internal clock to provide a Separated Clock output on pin 2. This output provides a window for the FDC to read the data. Separate short and long term counters in the 9216B ensure accurate clock separation. The Separated Clock output runs at 1/16 nominally of the internal clock. The divide ratio applied to the 4MHz clock is a function of the CD0 and CD1 inputs, pins 5 and 6. When the Osborne 4 is operating in the double density mode, both CD0 and CD1 are low, which creates a divide ratio of 1. This allows the internal clock to run at 4 MHz. Since the Separated Clock runs at 1/16 of this rate, the RDCLK to the FDC is 250 KHz in double density mode. When operating in single density mode, CD0 is high and CD1 is low. This creates a divide ratio of two, which results in an internal clock rate of 2 MHz and a RDCLK frequency of 125 KHz.

10.0 Introduction

The Osborne 4 has a seven-inch amber phosphor screen which can display a character matrix of 80 columns by 24 lines. The characters are stored in memory as 8-bit words. Seven of the bits define the standard ASCII code for the character, and one bit determines whether the character is to be displayed in inverse or normal video. Figure 10-1 shows how the display character is organized.

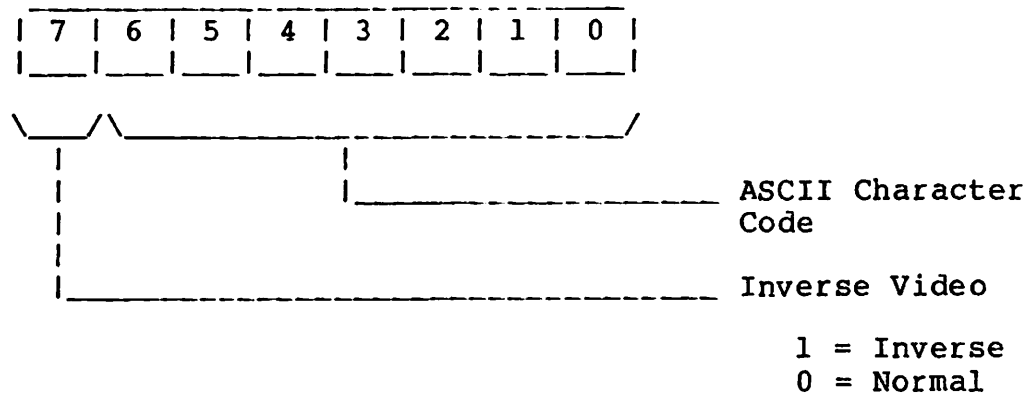


Figure 10-1. Video Display Character Organization

The characters that are to be displayed on the screen are stored in 3072 consecutive locations in main dynamic RAM, from F000H to FBFFH. These characters are placed in RAM as a result of reading keyboard key depressions, serial port data input, parallel port data input, etc. Figure 10-2 shows a partial memory map which illustrates the location of the memory-mapped video.

FFFF	Reserved
FC00 FBFF	
	Video Data
	80 Columns X 24 Rows
F000 EFFF	

Figure 10-2. Memory Map of Video Data

Section 10.0: Display Processor

When a line of text is to be displayed on the screen, these are the steps that are followed:

1. The software writes the data to be displayed into the appropriate addresses in main dynamic RAM. The data is written in standard ASCII format. For example, to write text onto the third line of the display, we would write into RAM locations F100H-F17FH.
2. Nothing happens with this data at F100H-F17FH until the display processor begins displaying the second line of text on the screen. The characters for this second line had previously been transferred from main dynamic RAM to a separate static RAM. This static RAM is organized as two 128-byte segments and is called the line buffer (a line meaning a line of text or 128 characters). As the display processor reads out the characters from one half of the line buffer to put those characters on the second line of the screen, it is also reading characters from main dynamic RAM for writing into the other 128-byte half of the line buffer for the third display line.
3. One half of the line buffer static RAM (128 bytes) now contains 128 characters ready to be put onto the third line of the screen. The other half of the buffer still contains the old characters that were used to form the second line. The display processor now transfers these characters from the line buffer to the third line on the screen while at the same time transferring new characters from main dynamic RAM to the other half of the line buffer, overwriting the old second line data. These new characters are for the fourth line, which will be displayed after the current third line is finished. This process continues for all 24 lines of text. The line buffer fills up with new data while emptying out previously loaded data to the screen.
4. Characters are read out of the line buffer one at a time and latched into a single-byte character buffer. The eight bits from the character buffer are used as address inputs to a character generator ROM. This ROM is programmed to output a unique set of bits for each character code at its address inputs. Each complete character consists of ten rows of eight horizontal dots resulting in each character occupying ten bytes of character generator ROM. The ROM uses its 4 most significant address lines, which are driven by a character scan counter, to determine which of the 10 bytes for that character is to be sent to the shift register. The CRT beam must scan across the screen 10 times in order to display a single row of characters.

Section 10.0: Display Processor

Since the beam must display the dots in a serial fashion across the screen, the 8 bits of parallel data out of the ROM are latched into a parallel to serial shift register and then clocked out a bit at a time by the 12 MHz Dot clock.

5. The serial bit stream, before display to the screen, is gated with an inverse video control line and a blank control line, to allow proper management of the screen.

The video processor maintains three sets of counters for accessing the appropriate characters for display.

1. The Character Address Counter
2. The Scan Counter
3. The Text Address Counter

Additionally, the Z80A provides an internal register used as a counter for refresh purposes used to set the lower 7 bits of the refresh address.

10.0.1 Character Address Counter

This is essentially a column counter that counts from 0 to 95 to position the characters properly from left to right across the screen.

10.0.2 Scan Counter

This counter is allowed to count once each time a complete dot row has been displayed. The scan counter counts from 0 to 9 then rolls over back to zero. The count from 0 to 9 corresponds to the scanning of the 10 rows that make up a single row of text on the screen.

10.0.3 Text Address Counter

This serves as the row counter, and it is incremented by one count each time the scan counter rolls over. This counter is responsible for counting from 0 to 23 to cause the 24 rows of text to be properly displayed from the top to the bottom of the screen.

10.1 Display Timing Control

The Dot clock, which runs at 12 MHz, is used to clock 5C located in area B4 on page 7. 5C counts up to 15, rolls over, and presets back to 8. So 5C continuously counts from 8 to 15. The only output of 5C used for timing is QC, which runs at 1.5 MHz. This output is inverted by 3E and

becomes the character clock (CHARCLK). CHARCLK is available at testpoint TP4. CHARCLK is used to run the three sets of counters on sheet 6 and to latch the character to be displayed in the character buffer register found on sheet 8. One cycle of this character clock corresponds to the time it takes to display one column on the screen.

10.2 Address Control

The display processor uses three sets of counters to keep track of the position of the character on the display. The column counter maintains a count of the 96 columns, the scan counter maintains a count of the 10 horizontal scans needed for each line of text, and the row counter maintains a count of the 24 rows of text.

10.2.1 Column Counter

The column counter is comprised of 4K and 4J on page 6, zones A4 and B4. This counter is cleared to zeroes by the CLRCHADR/ pulse from the display processor control ROM, 3J. It then counts up at the CHARCLK rate until it reaches a count of 5FH (95 decimal). At this point the 3J ROM clears the counter and the cycle repeats. So all this counter does is count from 0 to 95 (96 states total) under control of 3J.

The counter output lines, CHADR0-CHADR6, provide the addresses corresponding to each one of the 96 columns of the display. CHADR0-CHADR6 are used by the control ROM, 3J, to generate timing signals, and by the read address buffer on page 7 to read data from the line buffer into the character buffer.

10.2.2 Scan Address Counter

Each time the display control ROM, 3J, clears the column counter with a low going pulse, it presents a positive pulse through an inverter, 3E, to the count enable inputs of the scan counter 2J. 2J increments by one with this positive pulse each time a complete 128 column row is finished. The scan counter is disabled from counting during the time the column counter is actually counting because its enable inputs are held at a low state by the control ROM. The scan counter is an 74LS162 decade counter and counts from 0 to 9 and then rolls back over to zero. This cycle repeats continuously.

Section 10.0: Display Processor

The outputs of the scan counter, SCAN0-SCAN3, are used as the upper order address inputs to the character generator ROM, 2L, on page 8, zone C3. This ROM uses the scan count 0 to 9 to determine which of the 10 sets of 8 horizontal bit patterns to output for the particular character being created.

10.2.3 Row Address Counter

The row address counter consists of 2K, 5K, and 7K on sheet 6, zone D4. This counter is cleared by the CLRTXADR/ signal from the control ROM, 3J, after a full 24 lines of text have been output to the screen. The counter then counts up by one count each time the scan address counter rolls over. The row address counter counts from 0 to 23 and then is cleared by the control ROM. 2K provides four of the bits of the counter; the other bit is provided by gates 3J and 7K.

When the carry out of 2K occurs, a positive pulse one CHARCLK cycle wide appears at pin 15 and is fed to 5K pin 12. Up to this point, the other input to 5K, which is TXADR4, has been sitting at a low level, so the pin 11 output of 5K has been low all this time. The reason TXADR4 was low up until now is that when the CLRTXADR/ pulse occurred, it forced pin 11 of 7K to a low state, which was then clocked into 4L, a hex D flip-flop. The corresponding output of 4L (pin 7) was then set low. This is the TXADR4 signal. Now that the row address counter carry out has just occurred, pin 12 of 5K is high momentarily while pin 13 is low. This results in pin 11 going high. Now, pins 12 and 13 of 7K are both high, so pin 11 goes high and TXADR4 is latched to a high state by 4L. Since the lower 4 bits of the counter have just rolled over to zero, the row address counter is now sitting at a count of 10H. The ROM is monitoring this count and when it reaches 17H, (23 decimal) the ROM activates the CLRTXADR/ signal, which clears all five bits of the counter.

10.3 Control ROM

3J is the display processor control ROM; it is responsible for coordinating some of the most important timing and control signals used in the display processor. This 2732 ROM contains 4 Kbytes of storage. It uses the 7 address lines from the column counter and the 5 address lines from the row counter to generate its output signals at the proper times. Six signals are generated. Four of the six are latched by 4L. The ROM output data lines are not stable until after the address access time of the ROM is satisfied. This time is 450 nanoseconds for a 2732 ROM. The rising edge of CHARCLK causes the row and column counters to change state and present address information

to the ROM address inputs. Valid data appears 450 nanoseconds later at the outputs of the ROM. Since the period of CHRCLK is 666 nanoseconds, the data at the ROM outputs is stable long before the next rising edge of the clock. The next rising edge of CHARCLK is used to latch the ROM data into 4L and capture it just as the row and column counters are then clocked by the same rising edge and present new address information to the ROM inputs. The two signals that are not latched by 4L are used as synchronous clear inputs to the 74LS163 row and column counters. These clear inputs are not sampled by the 74LS163s until the rising edge of CHRCLK. They are handled by the 74LS163s in much the same manner that the ROM outputs are handled by 4L.

The timing signals provided by the control ROM are listed in Table 10-1.

Table 10-1. Control ROM (3J) Timing and Control Signals

CLRCHADR/	Clear Character Address
	This active low signal occurs after the column counter has reached 95. It is used to reset the column counter to zero.
HSYNC/	Horizontal Sync
	This active low output is used by the Osborne 4 CRT to do a retrace after each horizontal scan across the screen.
BLANK	Blank
	This signal is used to turn off the display during horizontal and vertical retracing.
COMP SYNC	Composite Sync
	This signal is used to provide horizontal sync pulses at the composite video output for use by a video monitor.
TXADR4	Text Address 4
	This is the most significant bit of the row counter. It is used as an address input to the control ROM.

Section 10.0: Display Processor

VSYNC

Vertical Sync

This signal creates vertical sync pulses for the Osborne 4 CRT. It is also used to interrupt the Z80A at a 60 Hz rate to invoke the keyboard scanning routine.

10.4 Line Buffer

The line buffer consists of two 128-byte segments of static RAM. One of the segments contains the 128 characters that are currently being transferred to a row of text on the screen, while the other segment is being updated from main dynamic RAM with characters that will be placed in the next row. The buffers change places at the end of each row of displayed text. Figure 10-4 portrays the two 128-byte segments and how they are exchanged between main RAM and the display.

Characters from RAM

Characters to Display

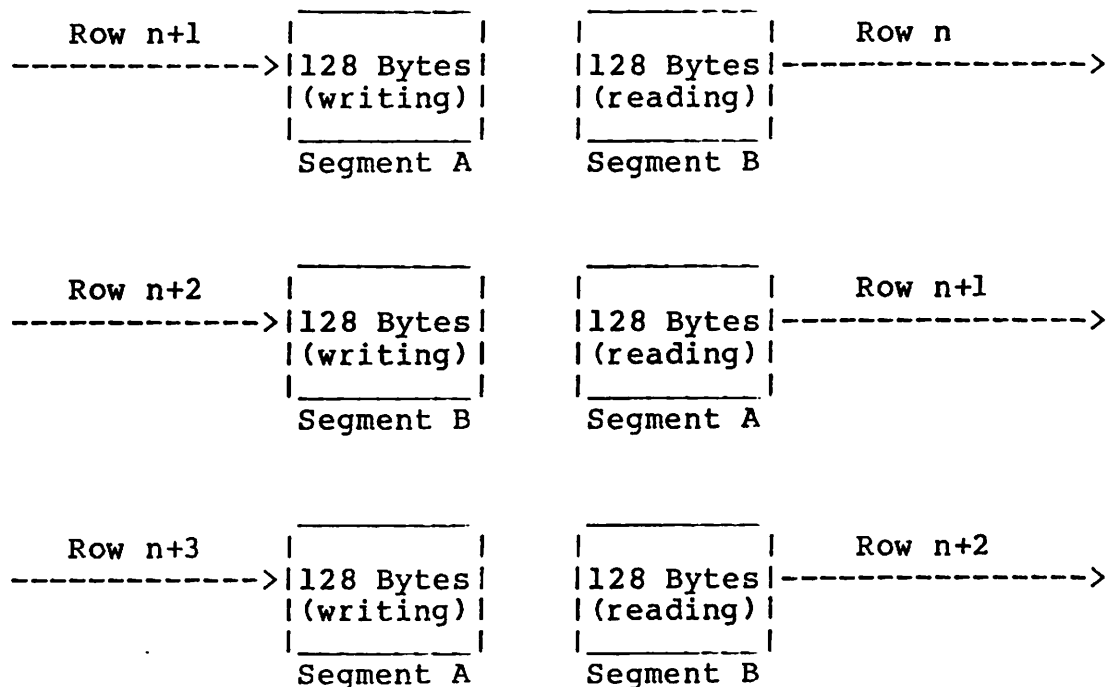


Figure 10-4. Management of Line Buffer RAM Segments

The line buffer is composed of two 1Kbyte X 4-bit, 2114 static RAM's, 2F and 2H, located at zone D1 and C1 of page 7. These RAMs provide a total of 1 Kbyte of storage, but only 256 bytes are actually used. Note that address inputs A8 and A9 of the RAMs are tied to ground, leaving 8 address lines active. This gives a total addressing range of 256 bytes.

10.4.1 Line Buffer Control

To understand how data gets in and out of the line buffer RAM, the line buffer control circuit must be examined. It consists of D flip-flop 2E and gates 7F and 6B. This circuit is responsible for creating the signals that control writing data to the line buffer, and those that control reading data from it to the character buffer.

Assume that both halves of 2E are initialized to a cleared condition. The circuit does nothing until the Z80A attempts a refresh cycle. While the Z80A thinks it is doing a refresh cycle on the 64 Kbyte RAM dynamic array, the RAM control circuitry is using the refresh address provided by the Z80A to read data out of dynamic RAM and write it to the line buffer under control of the line buffer control logic.

The data contained within the 64 K dynamic RAM chips needs to be refreshed periodically in order for the data to be maintained. The minimum refresh requirement is that 128 refresh cycles must be completed every two milliseconds. The Z80A has a built-in refresh mechanism that provides the RFSH/ signal along with MREQ/ during T3 and T4 of every instruction fetch cycle. The Z80A also has a 7-bit memory refresh register that it places onto its address lines during the refresh cycle, when RFSH/ and MREQ/ are both at a low level. This register increments from 0 to 127, under control of the Z80A, once for each refresh cycle. In this manner, the Z80A takes care of the RAM refresh requirement.

In most applications, a RAS/ only refresh is used. Instead of both the RAS/ and CAS/ memory signals being asserted, only the RAS/ signal is activated. This is to prevent the RAM from transferring any data during a refresh cycle. However, in the Osborne 4, data transfer does and must occur during "refresh" cycles. The data transfer referred to is that from main dynamic RAM to the line buffer.

As previously stated, the line buffer control circuit does nothing until a "refresh" cycle is initiated. RFSH, at pin 9 of 7F, goes high and sets 2E pin 2 to a high level. An active-low pulse then appears on the MREQ/ line and clocks 2E pin 5 to a high level on its rising edge. 2E pin 5 is fed back to the pin 10 input of 7F which holds 2E pin 2 at

Section 10.0: Display Processor

a high level. 2E pin 5 also goes to 2E pin 12 and causes 2E pin 9 to be clocked to a high level with the next rising edge of CHARCLK. 2E pin 9 stays high for one cycle of CHARCLK, then both pins 1 and 2 of 6B go high, forcing pin 3 low and resetting both halves of 2E. Nothing more happens until the next refresh cycle occurs.

When the line buffer control logic is activated by a refresh cycle, two important signals are produced:

1. Line Buffer Write Enable (LBWE/). This signal writes data into the line buffer RAM.
2. A strobe at 2E pin 5 that latches address and data from main RAM into 3F and 1F octal latches.

10.4.2 Writing to the Line Buffer

By examining the schematic, you will see that when 2E pin 5 goes high, it latches address and data from the main RAM into the Write Data and Write Address registers, 3F and 1F. The data bits are available to the line buffer as a result of the Z80A doing a refresh cycle to the main RAM. This refresh cycle actually causes data bits MDATA0-MDATA7 to be read out of the main RAM and written into the line buffer RAM.

The address being supplied to 3F consists of the 7-bit refresh count contained in the Z80A refresh register, which it places on the address lines during a refresh cycle.

The data character coming to the line buffer RAM from the main RAM has to come from the Video Data RAM area, which occupies memory from F000H to FBFFH. This means that the address going to the main RAM during a refresh cycle must lie within this range. All addresses going to main RAM must pass through a four-to-one address multiplexer. A 64 K dynamic RAM memory requires 16 address inputs, but these lines are presented to the RAM 8 bits at a time since the RAMs have multiplexed address inputs.

An examination of the four address multiplexers, 3A to 3D, on sheet 3 of the schematics shows that the multiplexers are controlled by two signals:

1. Column Enable (COLENB)
2. Refresh (RFSH/)

Section 10.0: Display Processor

During a refresh cycle, RFSH/ is low, and COLENB is just MREQ/ inverted. Initially, then, both control inputs to the multiplexers, pins 2 and 14, are low. This allows all the "0" multiplexer inputs to be sent to the RAM. When COLENB goes high (MREQ/ goes low), the "1" inputs are sent to the RAM. Each set of address bits is strobed by RAS/ and CAS/ into the RAM address input pins. RAS/ strobes in the 8 Least Significant Bits and CAS/ strobes in the 8 Most Significant Bits of address. Thus, a 16-bit address is stored inside the RAM. Figure 10-5 shows the address bit configuration that is input to the RAM during a refresh cycle.

RAM Address Bit	Signal	
A15	1	\
A14	1	
A13	1	
A12	1	/
A11	TXADR4	\
A10	TXADR3	
A9	TXADR2	
A8	TXADR1	
A7	TXADR0	/
A6	A6	\
A5	A5	
A4	A4	
A3	A3	
A2	A2	
A1	A1	
A0	A0	/

Pulled up to +5v

Row Address Counter Bits

Z80A Refresh Address Bits

Figure 10-5 RAM Address During Refresh Cycle

If the row counter and the refresh address bits are all zeroes, the RAM address during refresh is F000H. If the row address is at its maximum value (23 decimal) and the refresh address is at its maximum value (127 decimal), the refresh address is FBFFH.

The proper 128-byte area of data in main RAM is retrieved under control of the row address counter (TXADR0-TXADR6) and the Z80A refresh counter (A0-A6) and strobed into the write data register, 1F, on page 7. The refresh address (A0-A6) is also strobed into the write address register, 3F. These two registers are then enabled by the QC output of 5C and placed onto the line buffer RAM data and address lines. 2E pin 8 then provides a write pulse to store the data into the RAM. As the Z80A refresh address increments from 0 to 127, a line of text is transferred from main RAM to the line buffer RAM, and the main dynamic RAM is

Section 10.0: Display Processor

refreshed as well.

The data is written to either the high or low 128 byte segment of the line buffer RAM, depending on the state of address line A7 into the RAM. A7 is controlled by gate 5K, which has the least significant bit of row address, TXADR0, and CHARCLK/ as its inputs. When TXADR0 is low, data is written to the low segment, since TXADR0 is low and CHARCLK/ is high while the low segment RAM is written. When TXADR0 changes to a high level, A7 changes to a high while RAM is being written, so data gets stored in the high segment.

10.4.3 Reading From the Line Buffer

In order to make use of the data that is in the line buffer, it must be read out of the line buffer and into the character buffer. The Read Address Buffer, 3H, receives the column counter signals, CHADR0-CHADR6, and enables them onto the line buffer RAM address inputs during the time when CHARCLK is low. This is due to the fact that CHARCLK is connected to the 3H enable pins, 1 and 19. Note that while CHARCLK is high, Write Data and Write Address are enabled onto the line buffer data and address inputs, and while CHARCLK is low, the column counter address is enabled onto the address lines. When the line buffer RAM receives the column counter address, it outputs the data it has stored at that address onto the line buffer data lines, LBD0-LBD7. These signals carry 8 bits of data to the character buffer, 1H, at zone C4 of page 8. The data is latched into 1H on the rising edge of CHARCLK and held there until the next rising edge, at which point another character is loaded. Again, the 128 byte segment of the line buffer RAM from which the character is retrieved depends on the state of address line A7 into the RAM. A7 is controlled by TXADR0 and CHARCLK/. When TXADR0 is low, data is read from the high segment, since data is read while TXADR0 is low and CHARCLK/ is high. These conditions force A7 to a high level. When TXADR0 switches to a high level, A7 is forced low, and the data is read from the low segment. Table 10-2 summarizes the circumstances under which the line buffer RAM segments are written and read.

Table 10-2 Line Buffer RAM Segment Read/Write Control

CHARCLK	TXADR0	A7(5K-8)	Line Buffer Result
low	low	high	Read high segment
high	low	low	Write low segment
low	high	low	Read low segment
high	high	high	Write high segment

10.5 Character Buffer

The character buffer consists of an octal D flip-flop, 1H, on sheet 8 at zone C4. This buffer serves as temporary storage for the characters being read out of the line buffer. Characters are read out of the line buffer one at a time and latched into the character buffer on the rising edge of CHARCLK. Bits 0-6 of the character are presented to the character generator ROM directly while bit 7 is routed through a 2:1 multiplexer first. Bit 7 is used either as an inverse video bit for each character or as an extra bit to double character generation capability from 128 to 256 characters.

10.6 Character Generator ROM

The Osborne 4 is capable of generating 128 or 256 characters on the CRT screen, depending on the storage capacity of the character generator ROM, 1J. A 2716 ROM used here yields 128 characters, while a 2732 ROM gives 256 characters. You will recall that each character stored in the ROM actually occupies 10 bytes of storage, since the fully formed character space on the screen consists of 8 horizontal dots (one byte) by 10 vertical dots. Thus, 128 characters consumes 1280 bytes and 256 characters consumes 2560 bytes. A 2716 contains 2048 bytes and a 2732 contains 4096 bytes, so in both cases there is spare ROM storage space.

The character generator ROM uses the character buffer and scan counter outputs as its input address lines. The ROM simply does a lookup based on which of the 10 horizontal 8-bit patterns of a particular character need to be displayed at the moment.

The 8 bits coming from the character buffer can represent either of the following:

1. 7bitASCII (128 characters total) plus one bit for reverse video, or
2. 8 bit Character Set (256 characters total)

450 nanoseconds after the ROM is presented with an address, it places valid data on its output lines. This data is then clocked into a parallel to serial shift register, 2L, on the rising edge of ENBLDSR/, which is the pulse created when counter 5C on page 7 rolls over.

Section 10.0: Display Processor

As shown in figures 10-6 and 10-7, the top row of each character is stored in the first 128 (or 256) bytes of ROM. The second row is defined in the second 128 (256) bytes, and so on. Of the 128 ASCII characters, only the ones from 21H to 7EH are printable. The rest are control characters. See figure 10-8 for a complete chart of the ASCII character set.

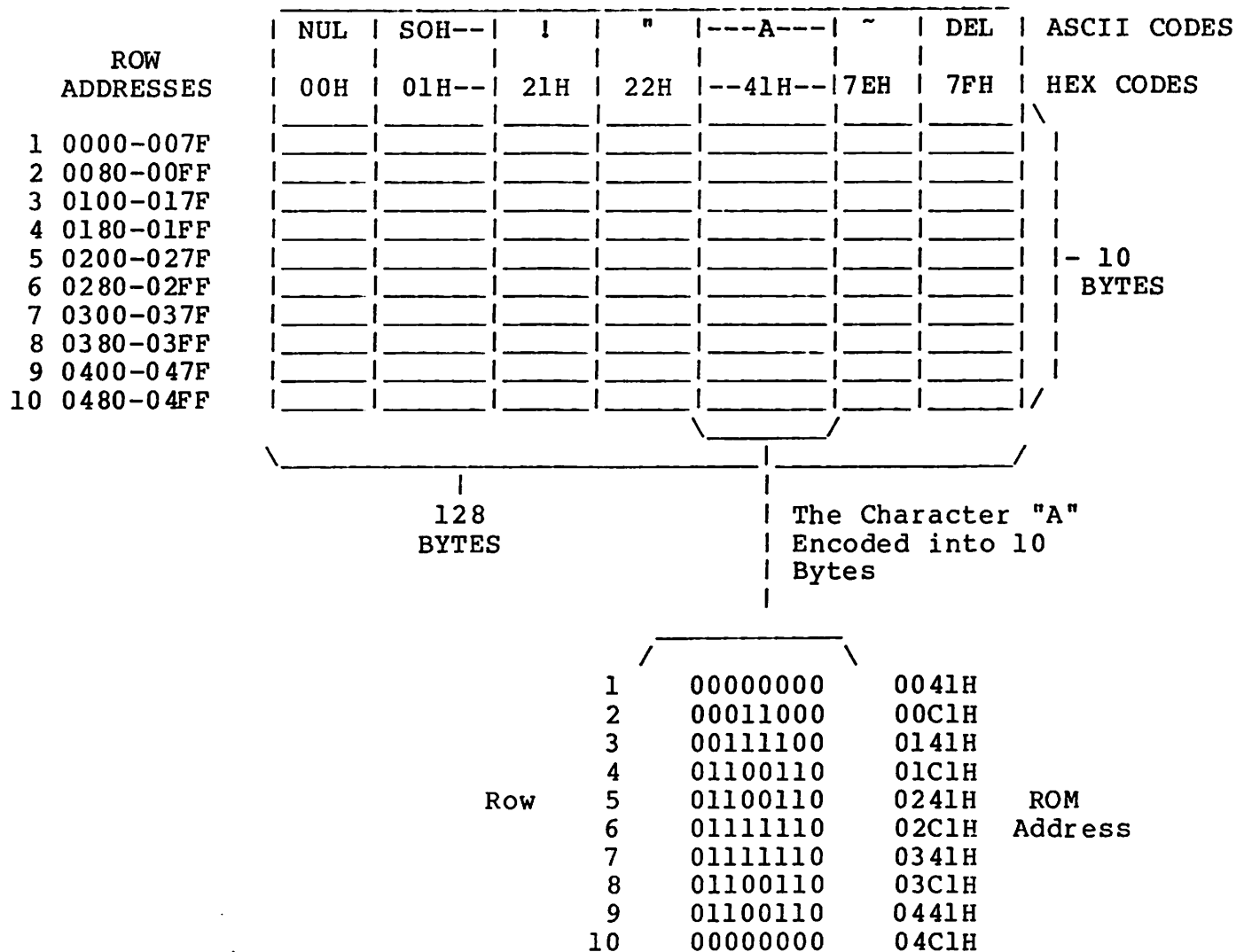


Figure 10-6. 128 Character Set ROM Organization

1. The first part of the document is a title page. It contains the title "THE HISTORY OF THE UNITED STATES OF AMERICA" and the author "BY JAMES MADISON".

1. The first part of the document is a title page. It contains the title "THE HISTORY OF THE UNITED STATES OF AMERICA" and the author "BY JAMES MADISON".

10

Section 10.0: Display Processor

00	NUL	22	"	44	D	66	f
01	SOH	23	#	45	E	67	g
02	STX	24	\$	46	F	68	h
03	ETX	25	%	47	G	69	i
04	EOT	26	&	48	H	6A	j
05	ENQ	27	'	49	I	6B	k
06	ACK	28	(	4A	J	6C	l
07	BEL	29	)	4B	K	6D	m
08	BS	2A	*	4C	L	6E	n
09	HT	2B	+	4D	M	6F	o
0A	LF	2C	,	4E	N	70	p
0B	VT	2D	-	4F	O	71	q
0C	FF	2E	.	50	P	72	r
0D	CR	2F	/	51	Q	73	s
0E	SO	30	0	52	R	74	t
0F	SI	31	1	53	S	75	u
10	DLE	32	2	54	T	76	v
11	DC1 (XON)	33	3	55	U	77	w
12	DC2 (TAPE)	34	4	56	V	78	x
13	DC3 (XOFF)	35	5	57	W	79	y
14	DC4 (TAPE)	36	6	58	X	7A	z
15	NAK	37	7	59	Y	7B	{
16	SYN	38	8	5A	Z	7C	
17	ETB	39	9	5B	[	7D	}
18	CAN	3A	:	5C	\	7E	~
19	EM	3B	;	5D	]	7F	DEL
1A	SUB	3C	<	5E	^		
1B	ESC	3D	=	5F	_		
1C	FS	3E	>	60	`		
1D	GS	3F	?	61	a		
1E	RS	40	@	62	b		
1F	US	41	A	63	c		
20	SP	42	B	64	d		
21	!	43	C	65	e		

Figure 10-8. Hexadecimal to ASCII Conversion

10.7 Output Video Control

The final video output control circuitry for Osborne 4 is all located on sheet 8 of the schematic. The following functions are provided:

- o A parallel to serial shift register
- o Reverse video
- o Alternate character sets
- o Video Monitor output
- o Composite Video output

10.7.1 Parallel to Serial Converter

After the character generator ROM outputs the 8-bit byte representing the 8 horizontal dots on the screen, the byte is latched into 2L, an 8-bit parallel-to-serial converter. The bits are then shifted out one at a time at the Dot clock rate of 12 Mhz. Just as the last bit is shifted out, 2L is loaded with a new byte. In this manner, a row of dots representing one tenth of a line of text is painted on the CRT.

10.7.2 Reverse Video

The video output control circuitry has the ability to create the reverse video attribute, either character by character, or for the entire video screen. Normal video is orange on a black background; reverse video is black on a orange background. 4N is the reverse video flip-flop. When its D input is high, the Q/ output, pin 8, goes low on the next rising edge of CHARCLK. Pin 8 is gated with the serial video data output and, when low, causes reverse video to be displayed. The input to 4N can be either continuously high, creating an entire screen of reverse video, or it can vary on a character to character basis, depending on multiplexer 4M. Output pin 7 of 4M controls reverse video. This output is multiplexed between bit 7 of the character buffer and the ALT CHAR SET/ signal. The select input of 4M is controlled by the 256 CHARS/ signal. 256 CHARS/ determines which of the above two mentioned signals gets output to 4M pin 7. When 256 CHARS/ is high, bit 7 of the character buffer is used as the reverse video bit. Since each character in a 128 character ASCII field has a spare bit for specifying reverse video, each individual character of this type can be displayed in either normal or reverse video. If bit 7 is high, reverse video is displayed.

Section 10.0: Display Processor

When 256 CHARS/ is low, the ALT CHAR SET/ signal is used to control reverse video. This cannot be controlled on a character by character basis, and so the entire screen carries either a normal or reverse video attribute. If ALT CHAR SET/ is high, reverse video results. Both the ALT CHAR SET/ and 256 CHARS/ signals originate from the 8155 PIO on sheet 4 of the schematic.

10.7.3 Alternate Character Sets

The Osborne 4 has the ability to display a 128 character set, a 256 character set, or two independent 128 character sets. This is done by using either a 2716 or 2732 ROM, and by controlling multiplexer 4M appropriately.

If a 2716 ROM is used, the only available option is a 128 character set. This character set is enabled only when both the ALT CHAR SET/ and 256 CHARS/ signals are at a high level. This allows the VPP input (pin 21) of the 2716 to go to a high state, allowing it to properly function.

If a 2732 ROM is used, the following options are available:

Option	Character Set	256 CHARS/	ALT CHAR SET/
1	256	0	0
2	256	0	1
3	128(low segment)	1	0
4	128(high segment)	1	1

Options 1 and 2 allow the All input of the character generator ROM to be determined by bit 7 of the character buffer. So, in this case 8 bits can be used to specify a character. This gives 256 combinations. If a full 256 characters are to be displayed, reverse video then cannot be enabled on an individual character basis because bit 7 has been used up to expand the character set.

Option 3 allows 128 characters stored in the lower segment of the ROM to be displayed, while option 4 allows the 128 characters in the upper segment to be displayed.

Table 10-3 summarizes the video output control options:

Table 10-3 Video Output Control Options

256 CHARS/	ALT CHAR SET/	2732 Characters	2716 Characters	Reverse Video
0	0	256	none	no
0	1	256	none	full screen
1	0	128 (lower)	none	character
1	1	128 (upper)	128	character

The standard Osborne 4 comes with a 256 character set contained in a 2732 ROM.

10.7.4 Video Monitor Output

The Osborne 4 CRT is controlled by signals from the video monitor connector, P2. The signals available at this interface are:

- o Video Out
- o Horizontal Sync
- o Vertical Sync
- o Brightness Control

10.7.4.1 Video Out

The video out signal consists of the video data from the parallel-to-serial converter 2L. This video out signal is gated with the reverse video and blank control signals. Blank is generated by the display control ROM on page 6 and, when high, forces 6K pin 6 to a high. This gets inverted by 7A and takes the video out signal at P2 pin 8 low, blanking the display. Blanking must be done during vertical and horizontal retracing on the CRT.

10.7.4.2 Horizontal Sync

This signal is directly output to the CRT from the video control ROM circuitry. This signal tells the CRT when to end its horizontal scan and start over at the next lower line on the display.

Section 10.0: Display Processor

10.7.4.3 Vertical Sync

This signal is directly output to the CRT from the video control ROM circuitry. It tells the CRT when to start over at the top of the screen. This occurs after the CRT has been completely scanned from top to bottom.

10.7.4.4 Brightness Control

Variable resistor R27 controls the CRT brightness.

10.7.5 Composite Video Output

The composite video output contains video and synchronization information all in the same signal. This is provided at the phono jack J4.

The composite video out signal is biased at approximately 3.8 volts by R5, R6, and R10. 7A pin 6 is an open-collector output, so when video is present, it is pulled up to the high state and allows the 3.8 volt output. When video is absent, 7A pin 5 goes to zero volts, so the video output drops to about 0.6 volts. When either a vertical or horizontal sync pulse occurs, 7A pin 4 goes to zero volts, and the composite video output goes all the way to zero volts.