Field Engineering Library

Theory of Operation

olivetti

PREFACE

The Olivetti L1 M24/M21 Theory of Operation Manual contains the description of the circuit boards present in the basic module of the M24 and M21 Personal Computers. It also contains the description of some of the optional boards that can be used with the M24/M21 Personal Computers.

This manual is intended for field engineers, system specialists and laboratory engineers who need to maintain and repair the boards present in the M24 and M21 Personal Computers.

DISTRIBUTION: Internal (Z)

FIRST EDITION: November 1984

REFERENCES:

M24/M21	- Hardware Architecture and Function	- code 4100710 W
M24	- Service Manual	- code 4100670 S
M21	- Service Manual	- code 4101680 C
M24	- Schematics	- code 4100750 S
M24	- Spare Parts Catalogue	- code 4100740 Z

TRADEMARKS:

IBM is a registered trade mark of International Business Machines Corp.

Intel is a registered trade mark of Intel Corp.

NEC is a registered trade mark of Nippon Electric Corp.

INS is a registered trade mark of National Semi-conductor Corp.

Microsoft and **MS-DOS** are registered trade marks of Microsoft Corp.

CP/M-86 is a registered trade mark of Digital Research.

Ethernet is a registered trade mark of Digital Research and Rank Xerox Corp.

OMNINET Transporter is a registered trade mark of Corvus Systems Inc.

Z8000 is a registered trade mark of Zilog Corp.

DTC is a registered trade mark of Data Technology Corp.

PUBLICATION ISSUED BY:

Ing. C. Olivetti & C., S.p.A
Direzione Documentazione
77, Via Jervis - 10015 IVREA (Italy)

CONTENTS

1. SYSTEM DESCRIPTION

CONTENTS

INTRODUCTION

This manual gives a detailed functional description of the circuit boards which reside in the Olivetti M24/M21 Personal Computers. It also gives the functional description of the keyboard and some optional boards that can be inserted into the system.

CHARACTERISTICS

Basically the M24/M21 Personal Computers are made up of the processing unit and the associated peripheral units (that is, disk drives, display unit, printers, keyboard). The processing unit and the peripheral control circuits reside in the Basic Module.

The Basic Module always contains the following circuit boards:

- Motherboard
- Indigenous Display Controller
- Power Supply

The basic module can be expanded by the addition of the following circuit boards:

- Bus Converter Board
- Memory Expansion Board
- Display Controller Options Board
- APB Z8000 Board

The last three boards can only be added to the system if the Bus Converter Board is present. In fact, they are plugged into the dual row connectors expansion slots of the Bus Converter Board.

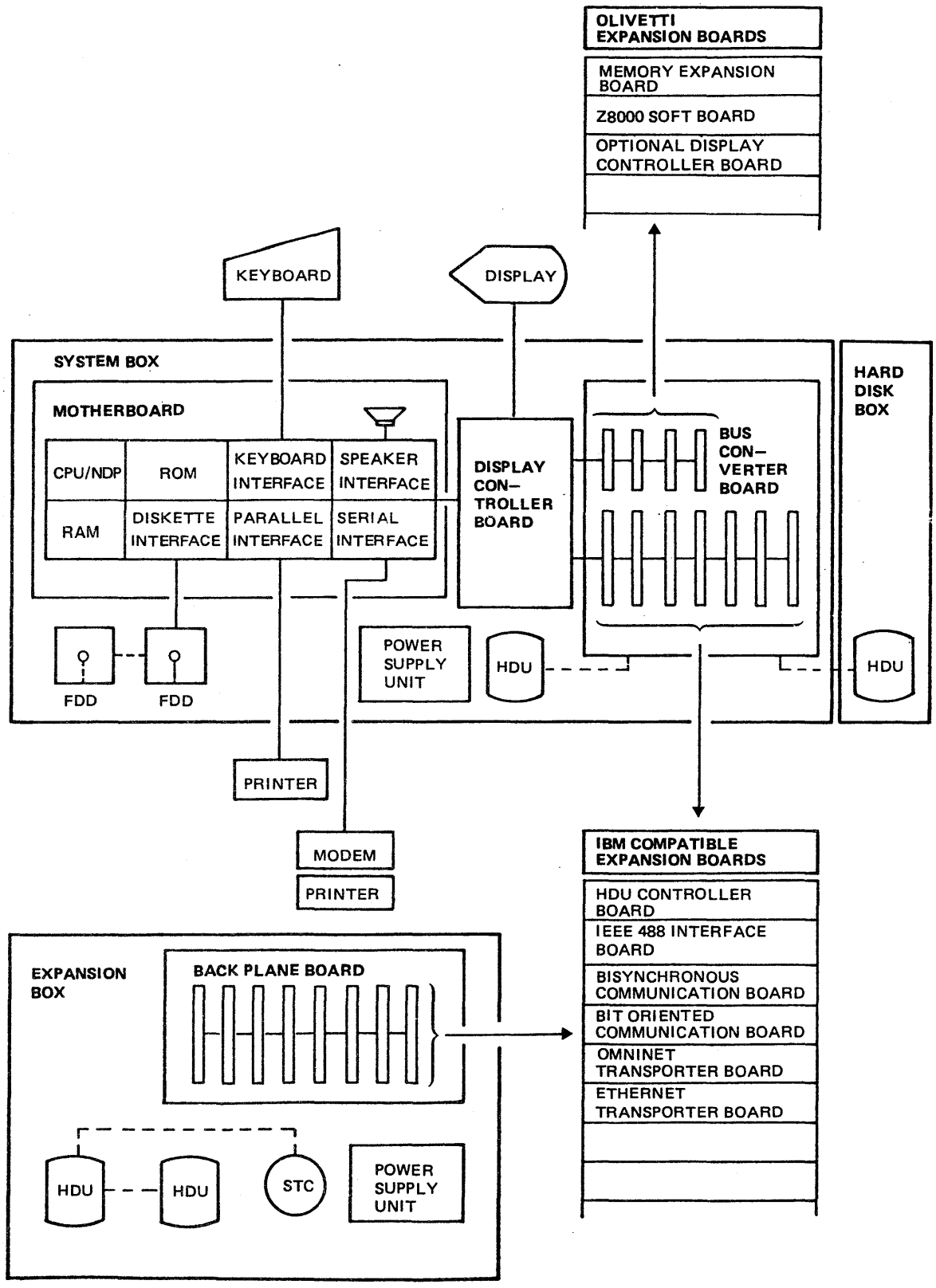


Fig. 1-1 System Block Diagram

With the Bus Converter Board present, other IBM compatible boards can be added to expand the system. These include:

- Hard Disk Controller
- Ethernet Transporter Board
- Ominet Transporter Board
- IEEE 488 Interface Board
- Twin RS232 Serial Interface Board
- Bisynchronous Communications Board
- Bit Oriented Communications Board
- Line Controller Unit Boards

These boards plug into the 62 pin connectors present on the Bus Converter Board.

CIRCUIT BOARDS

MOTHERBOARD

The motherboard is the main processing board of the M24/M21 systems. It is based on the 16 bit 8 MHz 8086 microprocessor and includes:

- **Central Processing Unit (CPU):** A 16 bit 8 MHz 8086 microprocessor which controls all the arithmetic and logic circuits within the system. It has a direct addressing capability up to 1 MB of memory, a 14 word by 16 bit register set and 24 operand addressing modes.

- **Numeric Data Processor (NDP) Socket:** This socket is available so that an 8087 Numeric Data processor can be installed as an option. This processor serves as a coprocessor attached to the 8086 CPU and effectively adds 80-bit floating point registers to the 8086 register set.

- **16KB of Read Only Memory (ROM):** Two 8K ROM chips which contain the Power up Diagnostics, the Power on Bootstrap, the drivers for mini-floppy disk drives, hard disk units and peripherals, and the initialization programs for Large Scale Integration chips.

- **Random Access Memory (RAM):** 128KB of RAM are available on the motherboard. These can be expanded up to 256KB with the insertion of 64K by 1 bit chips. When 256K by 1 bit chips are used the RAM can be expanded up to 512KB.

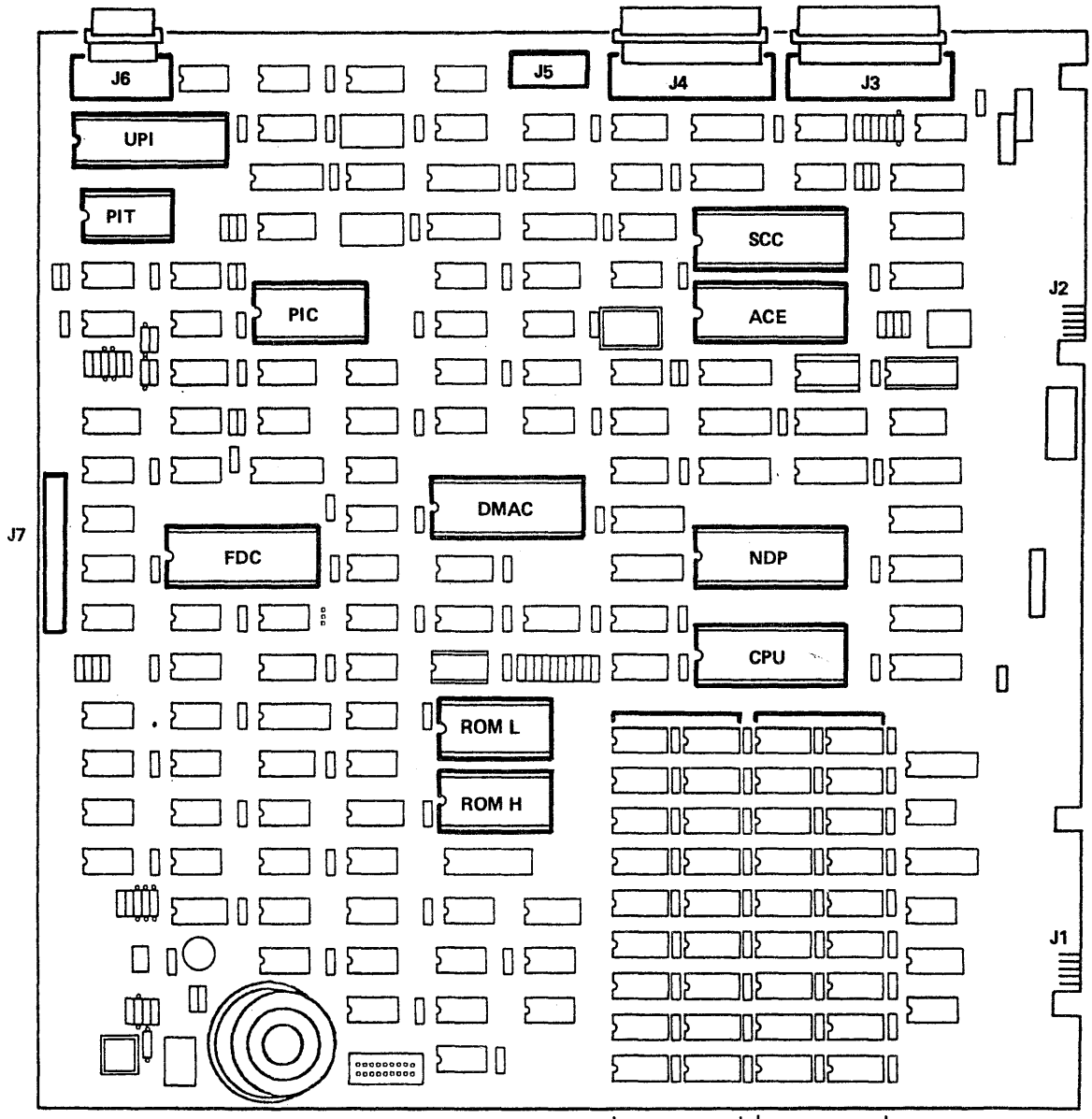
- **Memory Control Circuitry and Parity Checking:** This circuitry provides all the timing signals for the transfer of data to and from memory. It also multiplexes the addresses coming from the CPU and prepares them to address the particular memory location. The parity generation and checking circuitry generates a parity bit during a write operation and is then read back and checked for any parity error.

- **Interrupt Handling Circuitry:** It consists of an 8259 Interrupt Controller which is a programmable device that handles the priority interrupts to the CPU. It controls all the interrupts used in the system.

- **Bus Arbiter Logic:** The main component in this circuitry is the PAL 16R8 which handles the arbitration between the three bus masters namely the CPU, the DMA and an external processor.

- **DMA Circuitry:** This is based on the 8237 DMA controller and it allows the external I/O devices to transfer information to/from RAM memory directly without any CPU intervention.

- **Timer:** This is an 8253 programmable device used to provide a real time clock, to time and request refresh cycles from the DMA channel, and to provide the tone generator for the audio speaker.



CPU – 8086 CENTRAL PROCESSOR UNIT
NDP – 8087 NUMERIC DATA PROCESSOR
DMAC – 8237A–4 DMA CONTROLLER
ROM L – READ ONLY MEMORY LOW BYTE
ROM H – READ ONLY MEMORY HIGH BYTE
PIC – 8259 PROGRAMMABLE INTERRUPT CONTROLLER
PIT – 8253–5 PROGRAMMABLE INTERVAL TIMER
UPI – 8041A UNIVERSAL PERIPHERAL INTERFACE MICROCOMPUTER, KEYBOARD CONTROLLER
ACE – 8250 ASYNCHRONOUS COMMUNICATION ELEMENT
SCC – Z8530 SERIAL COMMUNICATION CONTROLLER
FDC – μPD765 FLOPPY DISK CONTROLLER

J1 – MOTHERBOARD TO DISPLAY
J2 – CONTROLLER BOARD INTERFACE
J3 – SERIAL INTERFACE
J4 – PARALLEL PRINTER INTERFACE
J5 – Z8350 CHANNEL B INTERFACE
J6 – KEYBOARD INTERFACE
J7 – DISKETTE DRIVE INTERFACE

Fig. 1-2 Motherboard

- **Clock Calendar Chip with Battery Back Up:** This chip is used to provide real time for the system. The battery back up is used to keep the device running even during a system power fail.

- **Parallel Interface:** This interface provides the system with one Centronics parallel port for connecting a printer.

- **Serial Interface:** The motherboard uses an 8250 asynchronous communication controller to provide an RS-232-C channel. It is possible to remove this chip and replace it with an 8530 Serial Communications Controller. The 8530 SCC is a USART (Universal Synchronous Asysnchronous Receiver Transmitter) and provides two serial channels.

- **Keyboard Interface:** This interface is based on an 8041 microprocessor which is used to convert the system parallel data to serial data for the keyboard and vice-versa.

- **Floppy Disk Interface:** The floppy disk interface is based on the uPD765 controller and it provides the circuitry required to drive and control two 5.25 inch mini-floppy disk drives.

INDIGENOUS DISPLAY CONTROLLER BOARD

The Indigenous Display Controller circuit board is a microprocessor based interface board for controlling colour and monochrome display units. It is based on a 6845 CRT controller and contains 32KB of RAM. It has a 4K or an 8 K byte ROM. With a 4KB ROM only the IBM character set is provided while with an 8KB ROM a second character set can be implemented.

The Display Controller can support the following modes of operation:

- 40x25 Alphanumeric Mode
- 80x25 Alphanumeric Mode
- 640x400 Monochrome Graphics Mode
- 640x200 IBM compatible Monochrome Graphics Mode
- 320x200 IBM compatible Colour Graphics Mode (with 12" Display Unit)
- 512x256, M20 compatible, Monochrome Graphics Mode (with APB Z8000 Board and 12" Display Unit present)

When in a colour mode, this controller can display four colours simultaneously, chosen from a palette of 16. With a monochrome monitor, instead of the various colours, shades of grey are displayed.

This Display Controller contains the circuitry for providing the following character attributes:

- Reverse Video
- Blinking
- Highlight
- Hide
- Underline

Other characteristics of this display controller include Degaussing and paging.

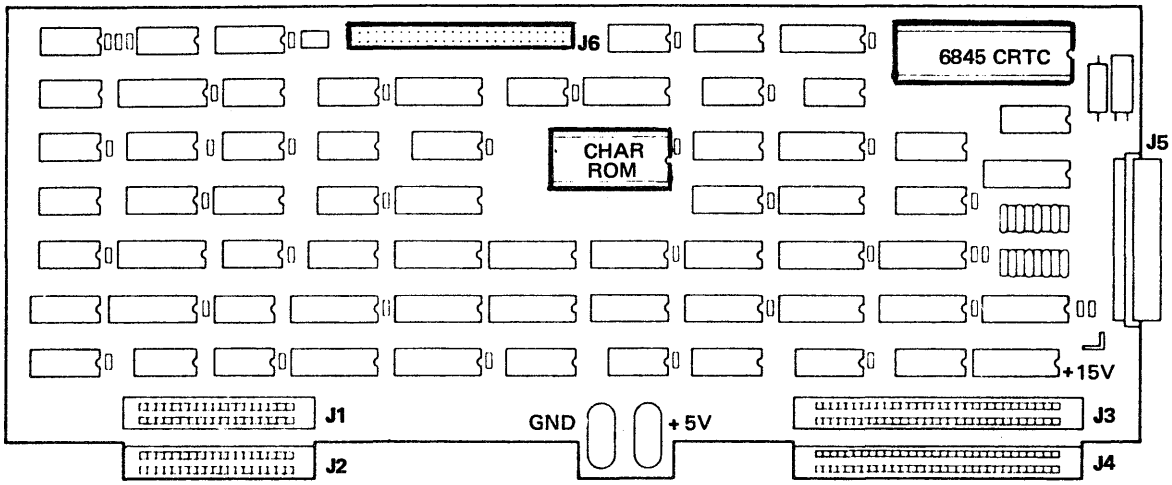


Fig. 1-3 Indigenous Display Controller Board

POWER SUPPLY BOARDS

The Power Supply Unit provides the correct d.c. voltages for all the circuit boards resident in the Basic Module, the keyboard, the disk drives and the monochrome display unit.

The circuitry lies on two boards enclosed in a metal case. It is a switching type power supply with a line input of 110Vac or 220Vac selectable by an on board jumper. The voltage outputs are full wave rectified and include the +5Vdc, +12Vdc, +15Vdc, -12Vdc voltages. The power supply circuitry also include overload/overvoltage protection for all voltages.

KEYBOARDS

Three different kinds of keyboards are used on the M24 and the M21 systems:

- Keyboard 1 - IBM compatible keyboard for M24 system
- Keyboard 2 - Native extended keyboard for M24 system
- M21 Keyboard - IBM compatible keyboard for M21 system

The three keyboards have the same circuitry but differ in their layouts and in the number of function keys.

The keyboard circuitry is based on the 8039 microprocessor which is used with a 16K byte EPROM. This EPROM contains the keyboard self-test routine, the scan codes for all keyboard keys and a routine to test for stuck keys. A 9 pin connector is also available to permit the connection of a "mouse" to the keyboard.

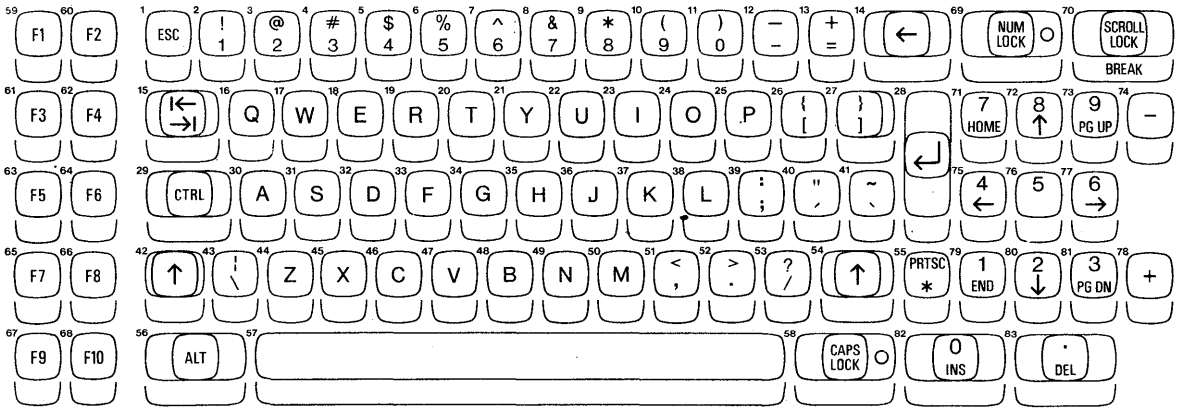


Fig. 1-4 Keyboard 1

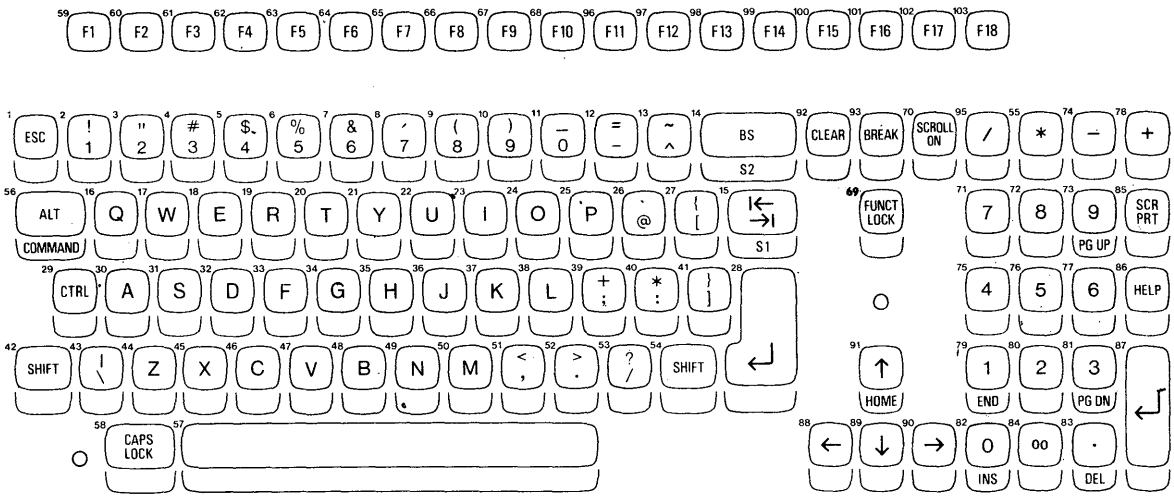


Fig. 1-5 Keyboard 2

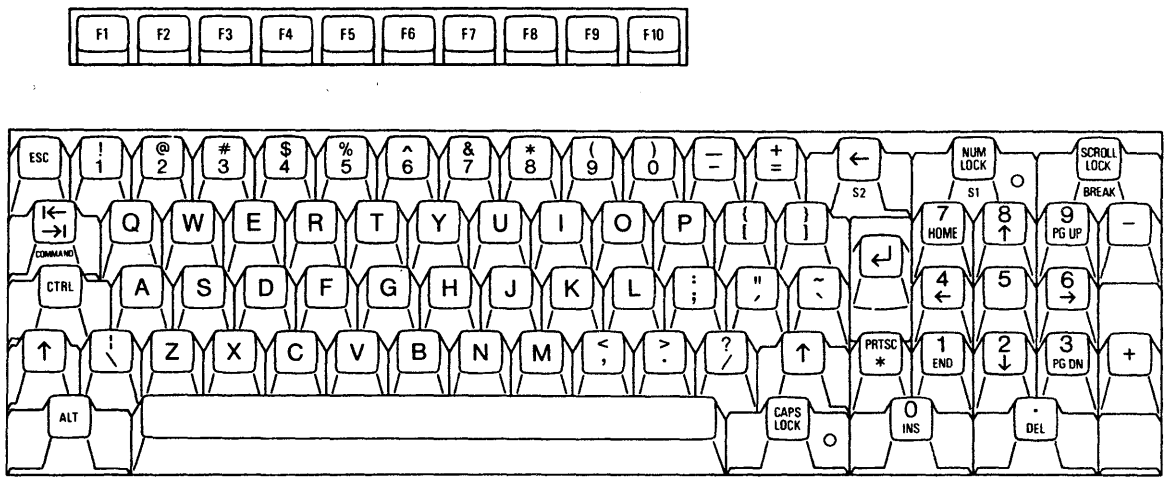


Fig. 1-6 M21 Keyboard

BUS CONVERTER BOARD

The Bus Converter Board is an optional board which when present provides 16 bit and 8 bit I/O expansion slots to the system. Thus it allows the simultaneous use of 16 bit Olivetti boards and 8 bit IBM compatible controllers.

It consists of an EPROM, addressed by a counter, which provides timing signals to enable and/or disable the required data buffers for I/O read or write operations to a controller.

This board contains two kinds of I/O bus connectors:

- 38 pin connectors to handle signals used by the 16 bit Olivetti boards.
- 62 pin connectors to handle signals used by both the 16 bit Olivetti boards and the 8 bit IBM compatible boards.

The M24 Bus Converter Board contains four 38 pin connectors and seven 62 pin connectors. The M21 Bus Converter Board contains one 38 pin connector and three 62 pin connectors.

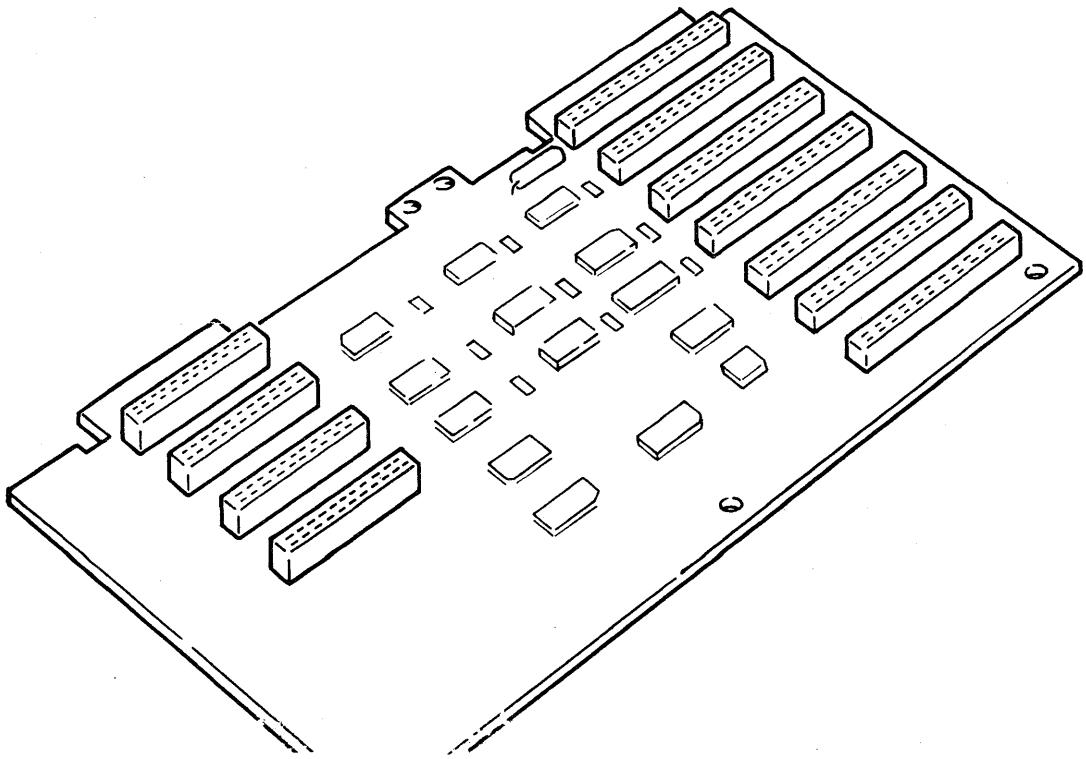


Fig. 1-7 M24 Bus Converter Board

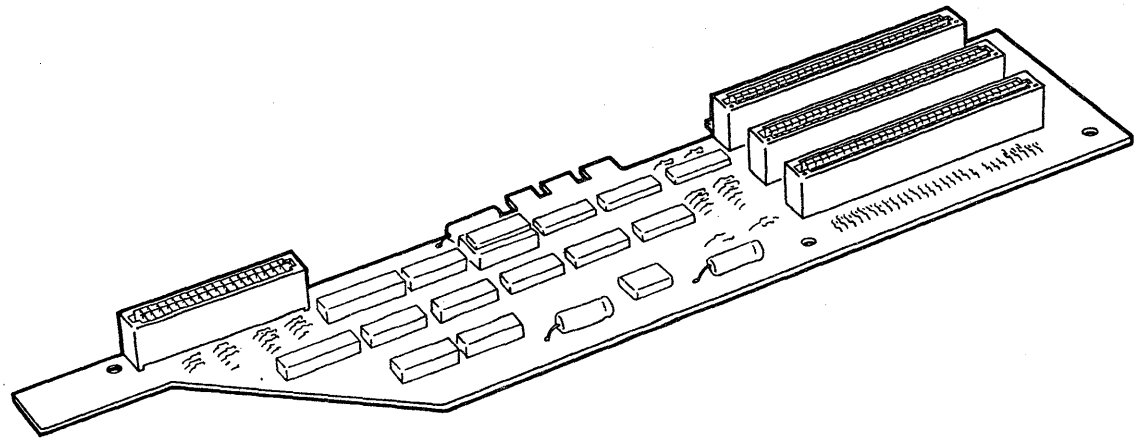


Fig. 1-8 M21 Bus Converter Board

MEMORY EXPANSION BOARD

The Memory Expansion Board allows the expansion of the system RAM memory by a maximum of 384KB. It consists of 3 banks, each of 128KB, and each bank contains 2 groups of nine 64Kx1 RAM chips.

Besides the three RAM banks, this board also performs the memory control logic, address multiplexing, bus buffering and parity generation and checking required for the added RAM memory.

It is inserted in a slot on the Bus Converter Board and it is seen by the system as a 16 bit board and thus it can perform byte and word operations.

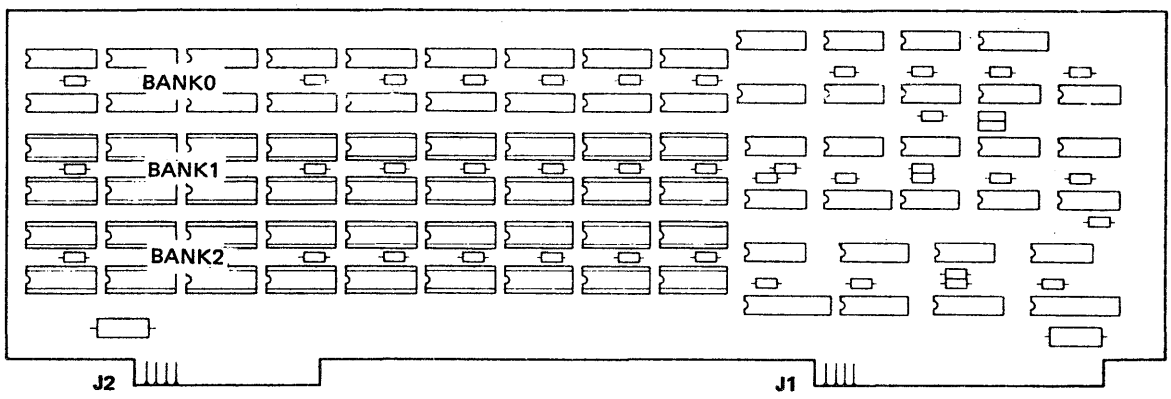


Fig. 1-9 Memory Expansion Board

APB Z8000 BOARD (M24 Only)

The APB Z8000 board is a 16 bit board which is inserted in the M24 Bus Converter Board. The function of this board is to allow the M24 system to operate in a PCOS (M20 operating system) environment instead of the M24 operating system. This permits M20 programs to be run on the M24 personal computer.

It is based on the Z8001 4MHz microprocessor and contains 8K bytes of resident ROMs used for bootstrap and diagnostic routines. It provides its own RS232 serial communication channel and uses the random access memory, the DMA circuitry and the peripheral controllers present on the motherboard.

The APB Z8000 board contains the bus handshaking circuitry which permits the system bus to be assigned either to the Z8001 processor or to the 8086 processor resident on the motherboard.

Also present on the APB Z8000 board is an address translation PROM which changes the Z8001 addresses to the physical memory addresses.

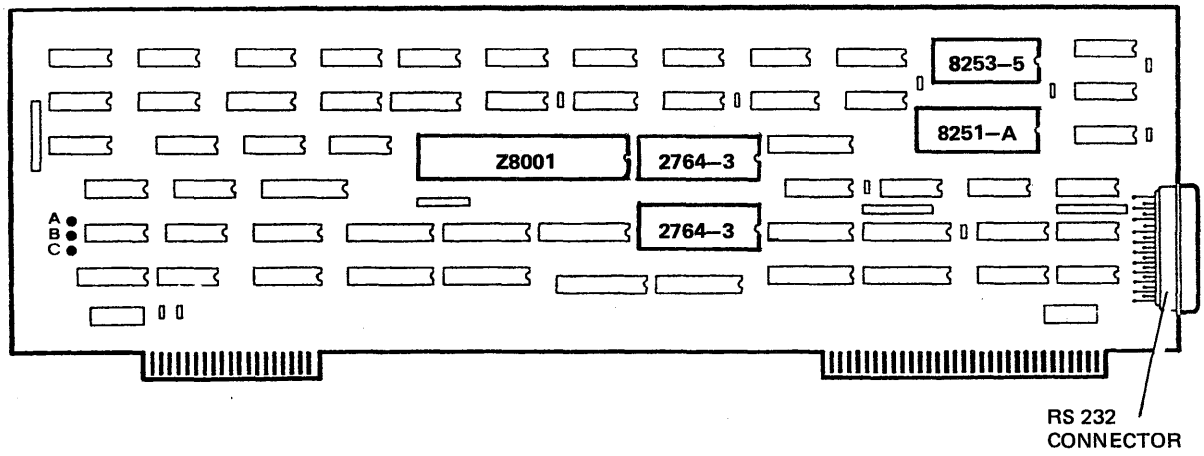


Fig. 1-10 APB Z8000 Board

DISPLAY CONTROLLER OPTIONS BOARD

The Display Controller Options Board is an optional board which is inserted into the dual expansion slots on the Bus Converter Board. Its use is to upgrade the Indigenous Display Controller Board to a high performance display controller. It is connected to the indigenous display controller via a small flat cable while the Bus Converter Board acts as an interface between the options board and the M24 system.

This board provides additional display features to the M24 Indigenous Display Controller which include:

- up to three additional 640 x 400 bit planes
- software controlled Look Up Table
- ability to display 16 colours simultaneously
- ability to display characters and graphics simultaneously
- blinking pixels in graphics modes
- 4 colour, 8 colour PCOS compatibility (with Z8000 softcard present)

The presence of this board in this system permits the connection of more than one monitor. In fact the enhancement board outputs go on to a 25 pin D-type connector on the board itself. When only one monitor is present, it must be connected to the connector on this board. The monitor can be either colour or monochrome. A second monitor can be connected to the indigenous board.

2. MOTHERBOARD

CONTENTS

INTRODUCTION

The motherboard is the large printed circuit board that sits at the bottom of the system box with the component side facing down. It consists of four major functional blocks.

- Processor/DMA Controller
- Memory (PROM and ROM)
- I/O peripheral interfaces
- I/O connectors

These functional blocks are connected by means of three system busses called A BUS, D BUS, and C BUS. Further busses are derived from these busses as explained later. Figure 2-1 shows the system Busses.

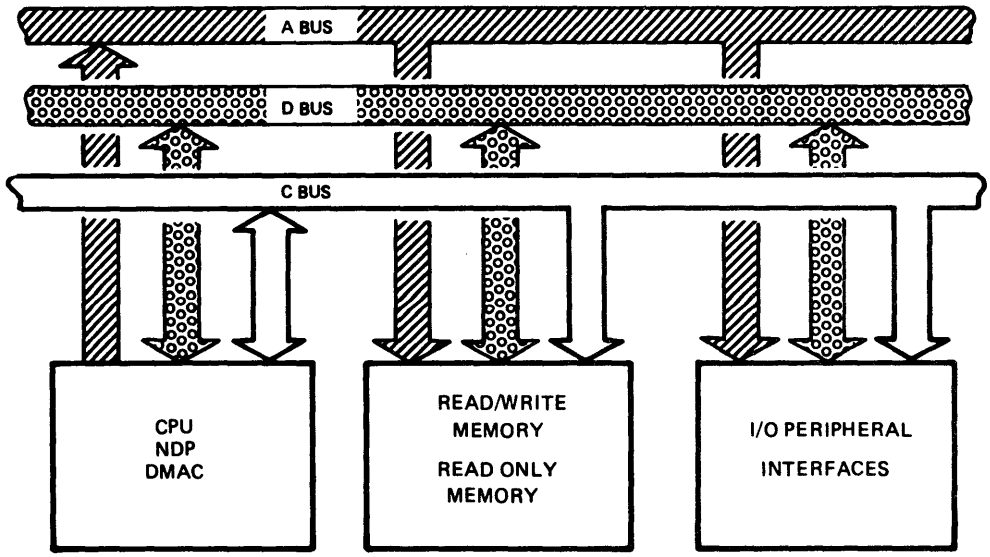


Fig. 2-1 System Busses

The A Bus carries the address information generated by the CPU, NDP or DMA Controller.

The following busses are derived from the A Bus:

- MA Bus which addresses RAM Memory and EPROMS
- BA Bus which addresses DMA Controller

The D Bus carries the data information between the processor/DMA controller, the memory and the I/O peripheral interfaces.

The following busses are derived from the D bus:

- MD Bus which carries data in and out of RAM memory and out of ROM memory

- PD Bus which carries data in and out of the DMA segment register, I/O peripheral interfaces and the Interrupt Controller.

The C Bus carries all the control signals.

MOTHERBOARD BLOCK DIAGRAM

Figure 2-2 is a detailed block diagram of the motherboard. Below is a list of the various modules and their function.

CENTRAL PROCESSOR UNIT (CPU)
16 bit microprocessor chip that contains within itself the arithmetic and logic circuits which extract program instructions from memory, one at a time and execute them. The microprocessor used is the Intel 8086 which runs at a frequency of 8 MHz.

NUMERIC DATA PROCESSOR (NDP)
16 bit numeric data processor that provides the instructions and data types needed for high performance. It is available as an option.

CLOCK GENERATOR
The clock generator is a single chip clock generator/driver which supplies the system clock for the CPU (8MHz) and a 4MHz clock for the peripherals. The clock generator chip used is the Intel 8284A.

CLOCK BUFFER
Circuitry which receives the system and peripheral clocks from the clock generator, buffers them and repowers them. Clocks generated include: CLK86 used mainly by CPU, NDP and Bus Controller; BCLK8 used by other logic on the motherboard; XCLK8 used by the I/O expansion boards. The integrated circuit used is a 74LS241.

BUS CONTROLLER
The Bus Controller utilizes the status outputs from the CPU to generate and buffer control signals. The chip used is the Intel 8288.

DMA CONTROL BUFFER
The LS245 bidirectional buffer serves two purposes. The first is to repower commands from the bus controller so that there is enough driving power for these signals to the I/O connector. The second is to repower the commands from the DMA Controller to the I/O connector when DMA is in control.

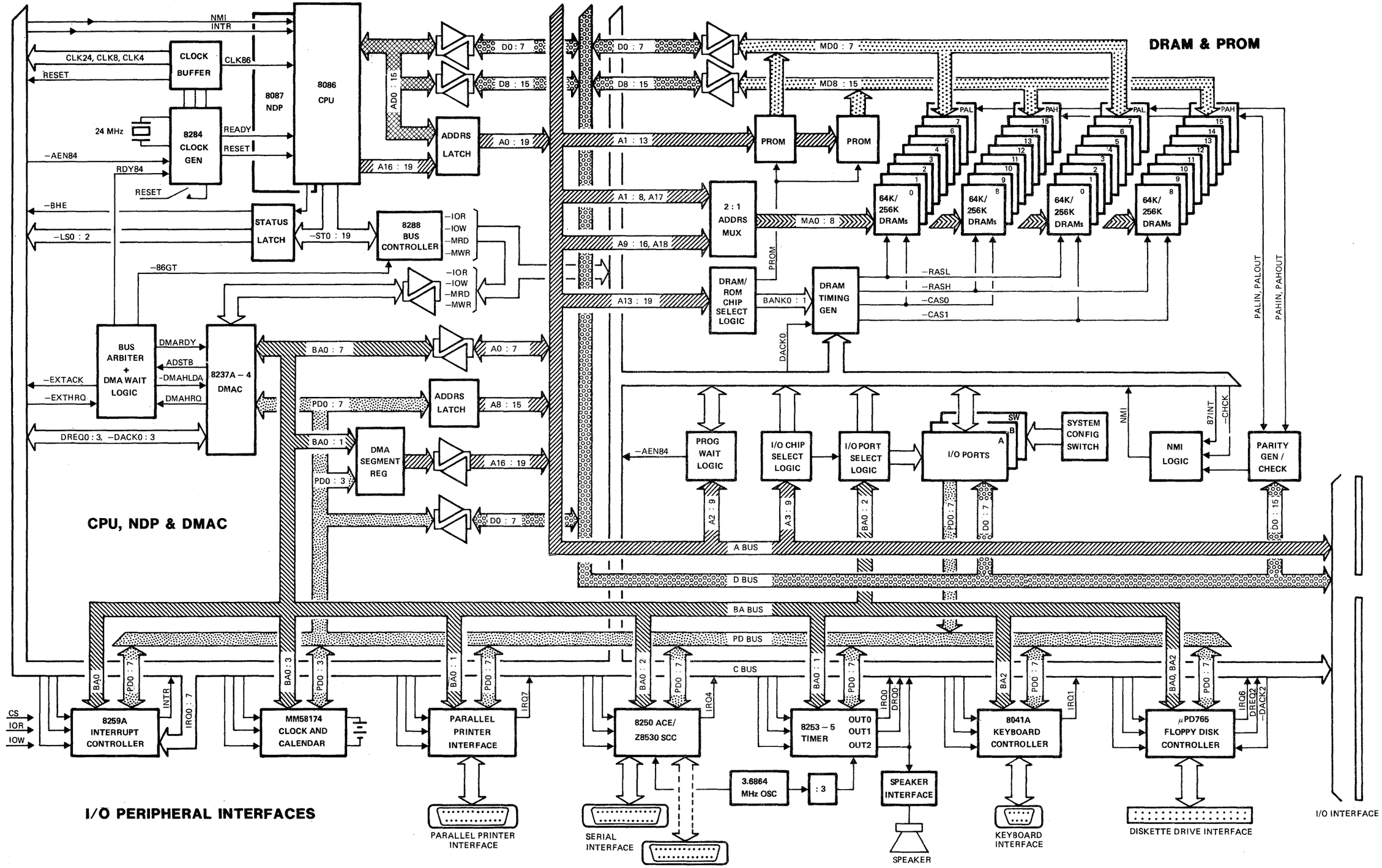


Fig. 2-2 Motherboard Block Diagram

CPU ADDRESS LATCHES
The CPU address is latched into the CPU Address Latches by the signal ALE (Address Latch Enable) from the Bus Controller. The Latch used is the 74LS373.

CPU DATA BUFFER
The CPU Data Buffer repowers the CPU data to the system D Bus. The buffer used is the bidirectional LS245.

BUS ARBITER CIRCUITRY
This circuitry arbitrates among the three system bus masters to access the system bus. The three system bus masters are: 8086 CPU, 8237 DMA Controller and an external processor (the Z8001 soft card for example). The main component of this circuitry is the Programmable Array Logic (PAL) 16R8 which handles this bus arbitration.

WAIT LOGIC
Each processor bus cycle consists of at least 4 clock cycles. These are referred to as T1, T2, T3 and T4. In the event of a 'NOT READY' indication being given by the addressed device, WAIT states (Tw) are inserted between T3 and T4. These WAIT states are generated by the programmable wait logic which mainly consists of a PROM and a counter. These circuits insert the exact number of WAIT states needed for all devices, except for DMA ones (for DMA devices one WAIT state is automatically inserted).

DMA CONTROLLER
Direct Memory Access Controller is the device which allows external I/O devices to transfer information directly to/from RAM memory. The integrated circuit used is the Intel 8237.

DMA SEGMENT REGISTER
A 4 x 4 74LS670 register file which provides four upper address bits (A16-A19) so as to permit the DMA Controller to address up to 1MB of memory.

DMA ADDRESS LATCHES
Latches used to interface the DMAC to the system A bus. Latches used are the 74LS373 and 74LS245.

RAM TIMING CIRCUITS
Circuitry which provides all the timing signals necessary to address memory and control the transfer of data or instructions to and from memory.

RAM ADDRESS MULTIPLEXER
Circuitry made up of multiplexers used to address RAM. The system memory can be addressed in bytes as well as in words addressing two bytes. The multiplexer used is the 74S158. This 2:1 multiplexer is used to switch either row address or column address to RAM at a certain required time.

RANDOM ACCESS MEMORY
Volatile memory which stores the Operating System, Interpreter, and all other user data and programs. Information stored in RAM may be altered. This Personal Computer uses 64K by 1 bit dynamic RAMs or 256K by 1 bit dynamic RAMs.

RAM PARITY GENERATOR AND CHECKER
Circuitry which generates the parity bit that gets written into memory during memory write operations. Parity is read back and checked against data to see if there is any parity error. If any parity error is detected, the parity error flip-flop is set and a CPU Non-Maskable Interrupt is generated. NMI is reserved for some event, as in this case, that cannot wait.

ERASABLE PROGRAMMABLE READ ONLY MEMORY (EPROM)
EPROM is used to store the power up diagnostics and bootstrap. It is a non-volatile type of memory and its capacity is 16KB

INPUT/OUTPUT CHIP SELECT LOGIC
Circuitry used as motherboard I/O address decoder. A PAL12L10 is used as this decoder.

MINI-FLOPPY DISK CONTROLLER
The circuitry which provides all the logic and control necessary to control and record data onto, or read from the 5.25 inch mini-floppy disks. It also initially formats new disks. The LSI component used is the uPD765.

TIMER
Programmable device used to provide a real time clock, to time and request refresh cycles for DMA channel and to provide the tone generator for the audio speaker.

INTERRUPT CONTROL LOGIC
Programmable device that handles the priority interrupts to the CPU. It functions as an overall manager in the interrupt-driven system environment. The LSI component used is the Intel 8259.

CLOCK CALENDAR CHIP
A battery backed-up clock calendar chip which provides real time such as seconds, minutes, hours, day of week, days and months for the system.

SERIAL COMMUNICATION CONTROLLER
The 8250 Asynchronous Communication Element is used to support serial communication and is configured to have an RS232-C channel. The 8250 ACE can be replaced by the Z8530 SCC to have an asynchronous/ synchronous communication channel.

PARALLEL PRINTER INTERFACE
Centronics-Like Parallel Printer Interface used to interface the personal computer to one of the parallel printers.

KEYBOARD CONTROLLER
Circuitry that converts the system parallel data to serial data for the keyboard and vice-versa. The LSI component used is the Intel 8041.

CENTRAL PROCESSOR UNIT

The central processor unit is the heart of the motherboard. Its function is to extract program instructions from memory and execute them. It consists of an Intel 8086 microprocessor chip, which contains internally the arithmetic and logic circuits required for executing the program instructions resident in memory.

The system can also use a Numeric Data Processor (NDP) which acts as a co-processor to the CPU. It provides other mathematical instructions for all data types needed for high performance computing. The NDP is an Intel 8087 chip and is an optional feature.

Externally, there are additional logic address decoding, timing and buffer elements which are necessary to address memory and to control data and instructions.

CPU PIN FUNCTIONS

Figure 2-3 shows the pin functions for the Intel 8086 microprocessor in maximum mode and a brief description of each pin follows:

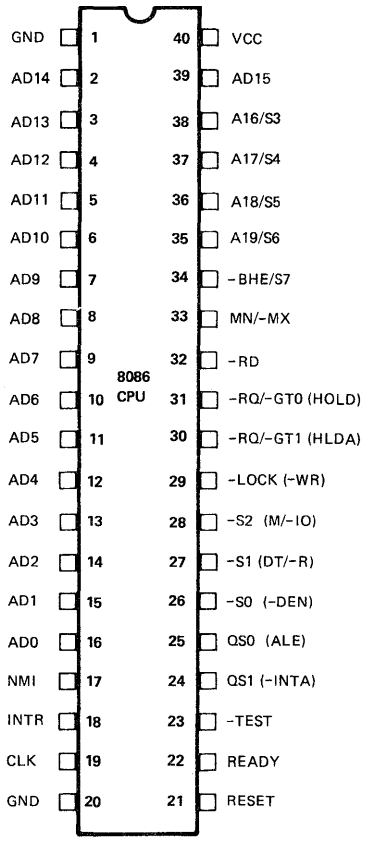


Fig. 2-3 8086 Pin Functions

AD0-AD15 Address/Data Bus (Input/Output, active high)
These lines constitute the multiplexed memory/IO address and data bus. A0 is used to enable data onto the least significant half of the data bus, D0-D7, in memory or I/O operations.

A16/S3-A19/S6 Address/Status (Output, active high).
During T1 these are the four most significant address lines for memory operations. During memory and I/O operations, status information is available on these lines during T2,T3,Tw and T4. During I/O operations these status lines are low.

MN/-MX Minimum/Maximum (Input)
This is the input used to select minimum or maximum mode for the 8086. In this case, maximum mode is used to support the 8087 NDP.

-BHE/S7 Bus High Enable/Status (Output, active low)
During T1 the -BHE signal is used to enable data onto the most significant half of the data bus D8-D15. The S7 status information is available during T2,T3, and T4.

-S2 to -S0 Status Lines (Output, active low)
These are the status lines for memory/IO transactions, interrupt acknowledge, processor halt or passive state.

-S2	-S1	-S0	Function
0	0	0	Interrupt Acknowledge
0	0	1	Read I/O port
0	1	0	Write I/O Port
0	1	1	Halt
1	0	0	Code Address
1	0	1	Read Memory
1	1	0	Write Memory
1	1	1	Passive

QS0-QS1 Queue Status (Output)
These lines are the status lines for the 8086 internal instruction queue.

-TEST Test (Input)
This input is examined by the "wait" instruction. If it is low, execution continues, otherwise the processor waits in an "idle" state.

READY Ready (Input, active high)
This pin is used by addressed memory or I/O device to insert the required number of wait cycles. The READY signal from memory/IO is synchronized by the 8284A Clock Generator to form READY.

-RQ/-GT0, -RQ/-GT1 Request/Grant (Input/Output)
These lines are used by other local bus masters to force the processor to release the local bus at the end of the processor's current bus cycle. -RQ/-GT0 has higher priority than -RQ/-GT1

NMI Non-Maskable Interrupt (Input)
This is an edge triggered input which causes a type 2 interrupt. A transition from a low to a high initiates the interrupt at the end of the current instruction. The NMI interrupt has the highest priority over the other interrupts.

INTR Interrupt Request (Input, active high)
This line is a level triggered input which is sampled during the last clock cycle of each instruction to determine if the processor should enter into an interrupt acknowledge operation.

-LOCK Lock (Output, active low)
This is to inform other system bus masters that the system bus won't be released while this pin is active.

-RD READ (Output)
This read strobe indicates that the processor is performing a memory or I/O read cycle. This is not used in the this system.

RESET Reset (Input, active high)
This causes the processor to terminate its present activity immediately.

CLK Clock (Input)
This provides the basic timing for the processor and bus controller.

GENERAL OPERATION

The internal functions of the Intel 8086 microprocessor are divided into two major functional units

- Execution/Control Unit (EU)

- Bus Interface Unit (BIU)

As shown in figure 2-4, these units can interact directly but for the most part perform their functions separately.

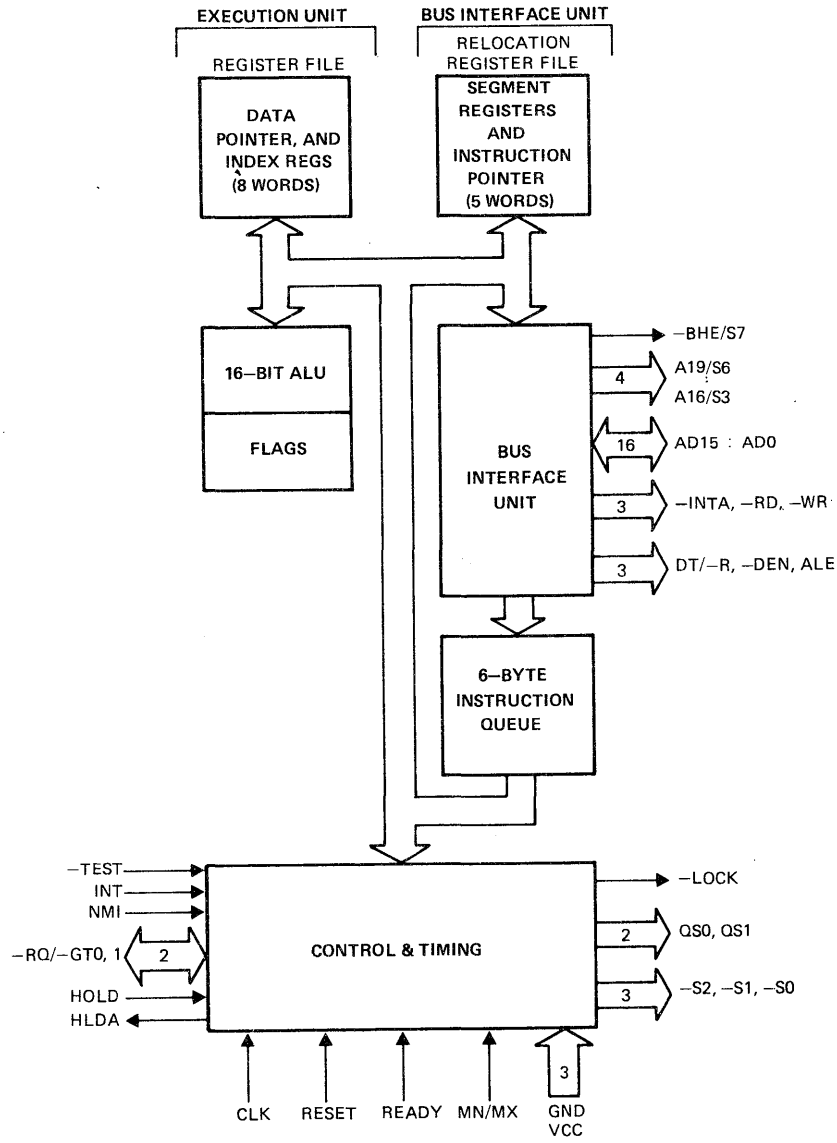


Fig. 2-4 8086 Functional Block Diagram

The execution unit (EU) performs the basic processing functions, as it contains the data registers and the arithmetic logic unit (ALU). It accepts prefetched instructions from the BIU and returns unrelocated operand addresses to it. It then receives memory operands via the BIU, processes them, and passes the results to the BIU for storage.

The Bus Interface Unit (BIU) first prefetches instructions before they are required by the EU. It buffers them in a queue that can contain up to six bytes of instruction stream, to wait for decoding and execution. The EU therefore does not need to wait for completion of a bus cycle before taking in a new instruction. The BIU also provides the functions related to operand fetch and store, address relocation, and bus control processing.

CPU - MEMORY OPERATION

The 8086 processor employs a 20 bit address bus to access a byte or word in memory. The memory is logically organized as a linear array of 1 Mbyte addressed as 00000 to FFFFF (hex). Each location is an 8 bit byte.

Word (16 bit) operands consisting of consecutive bytes can fall on either even or odd address boundaries. The processor provides two signals, -BHE and A0, to select and enable an odd location, an even location or both.

For address and data operands, the least significant byte of the word will be stored in the lower valued address location and the most significant byte in the next highest address location.

The BIU automatically performs the proper number of memory accesses, one if the word operand is on an even byte boundary and two if it is on an odd byte boundary.

Physically the memory is organized as a high bank (D15-D8) and a low bank (D7-D0) which are addressed in parallel by the processor's address lines A19-A1. Byte data with even addresses is transferred on the D7 - D0 bus lines while odd addressed byte data (A0 high) is transferred on the D15-D8 bus lines.

-BHE	A0	
0	0	whole word
0	1	upper byte from/to odd address
1	0	lower byte from/to even address
1	1	none

CPU BUS OPERATION

The 8086 microprocessor has a combined address and data bus called a time multiplexed bus. This bus is demultiplexed at the processor with a single set of address latches.

Each processor bus cycle consists of at least four CLK cycles referred to as T1,T2,T3 and T4. The address is emitted from the processor during T1 and transfer occurs on the bus during T3 and T4.

T2 is used for changing the direction during read operations. In the event of a "NOT READY" indication being given by the addressed device, "WAIT" states (Tw) are inserted between T3 and T4. Periods can occur between 8086 driven bus cycles referred to as "Idle" states. The processor uses them for internal housekeeping. During T1 of any bus cycle the ALE (Addressing Latch Enable) signal is emitted by the bus controller. At the trailing edge of this pulse, a valid address and certain status information for the cycle may be latched. Status bits -S0 to -S2 are used by the bus controller to identify the type of bus transaction.

CPU I/O OPERATION

The 8086 microprocessor provides 64K addressable input or output ports. I/O space is addressable as if it were a single memory segment, without the use of segment registers. The I/O address appears in the same format as the memory address on bus lines A15-A0.

NUMERIC DATA PROCESSOR

The 8087 Numeric Data Processor serves as a coprocessor attached to the 8086 CPU. It effectively adds eight 80-bit floating point registers to the 8086 register set. It uses its own instruction queue to monitor the 8086 instruction stream, executing only those instructions intended for it and ignoring the instructions needed for the 8086 CPU. The 8087 requires the same type of timing, power and bus structure as the 8086 in maximum mode. The 8087 NDP instructions include a full set of arithmetic functions as well as powerful exponential, logarithmic and trigonometric functions.

The 8087 NDP cannot run by itself as it needs the 8086 CPU to run the data, address and control busses which feed it instructions and operands. Figure 2-5 shows how the NDP is attached to the 8086 CPU. There are several lines running directly between the NDP and the CPU, namely:

- The test-busy signal
- A request/grant (RQ/-GT0) line
- Queue status (QS1, QS0) signals

The test input pin of the 8086 is connected to the BUSY output pin of the NDP. This allows the 8086 CPU to use the WAIT instruction before each NDP instruction and for the programmer to put an FWAIT instruction in the program following each NDP instruction which deposits data in memory for immediate use by the CPU. Then the numeric instruction gets translated to the indicated NDP numeric operation (with the preceding WAIT) and the FWAIT instruction is translated as the CPU WAIT instruction. While the 8087 NDP is executing a numeric operation, it puts a 1 on its busy pin (hence the test pin of the CPU is forced to a 1). While the 8086 CPU executes a WAIT instruction, it halts its activity until the test pin (pin 23) is returned to its normal state(0).

Thus the sequence of an NDP numeric instruction followed by a CPU WAIT will cause the CPU to call the NDP and then wait until the NDP has finished before proceeding.

The request/grant line RQ/-GT0 is used by the NDP to gain control of the bus which is shared by the NDP and CPU. This request/grant on the NDP line is connected to the RQ/-GT1 of the 8086 CPU. This is a two way communication line. A signal (request) from the NDP to the CPU indicates that the NDP wants to use the bus. Before the NDP can take the bus it must wait for the return signal (grant) from the CPU. When the NDP finishes with the bus it sends a signal back to the CPU on the same line to indicate this. Thus, it is the NDP which requests the bus, and the CPU is the device which grants the bus as soon as it can after such a request.

There are two queue pins, QS1 and QS0, which help the NDP to keep its instruction queue synchronized with the CPU's. The two bits are used to encode the four possible states:

QS1	QS0	Function
0	0	No operation
0	1	First Byte of instruction
1	0	Empty the queue
1	1	Subsequent byte of an instruction

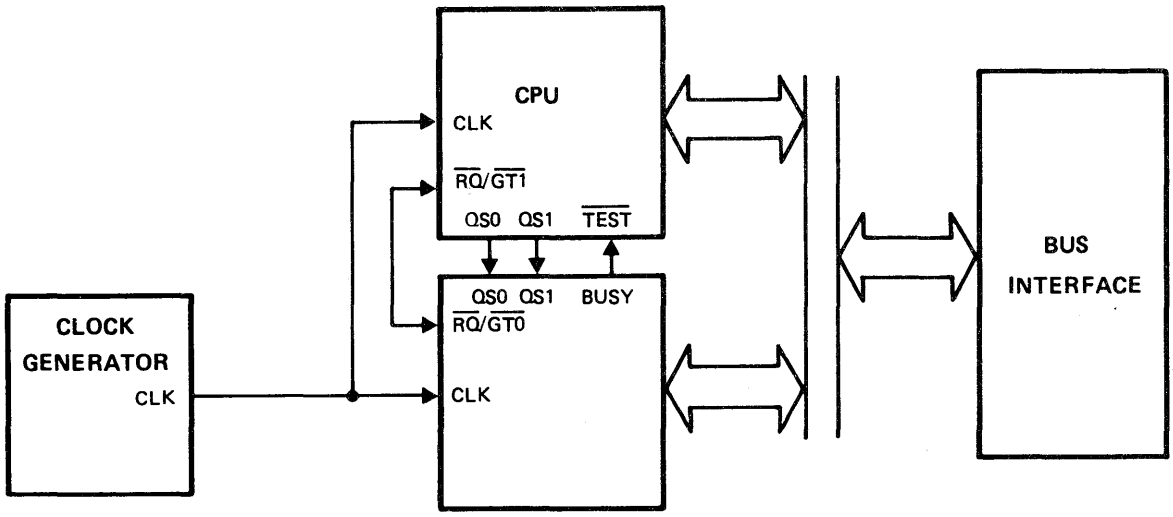


Fig. 2-5 8087 NDP - 8086 CPU Connection

DMA CONTROL LOGIC

DMA stands for Direct Memory Access and the main component of the DMA circuitry is the Intel 8237 DMA Controller. The DMA Controller is the device which takes over the system bus to transfer information directly from the I/O devices to the system memory and vice versa. This is necessary because blocks of data often have to be moved very rapidly.

The DMA circuitry is mainly made up of:

- DMA Controller
- DMA Segment Register
- DMA Control Buffer
- DMA Data Buffer
- DMA Address Latches

A simple DMA transfer usually takes place as follows: The DMA Controller is told to make a transfer either by the CPU or other device; then the DMA Controller makes a request to gain control of the bus from the CPU, other processors, or controllers which might currently be using the bus; these other devices then relinquish control of the bus by putting their lines into the tri-state condition (electrically disconnecting the lines); they then grant the bus to the DMA Controller. Finally, the DMA Controller takes over the bus, generating its own address and control signals for the bus and causing the transfer of information.

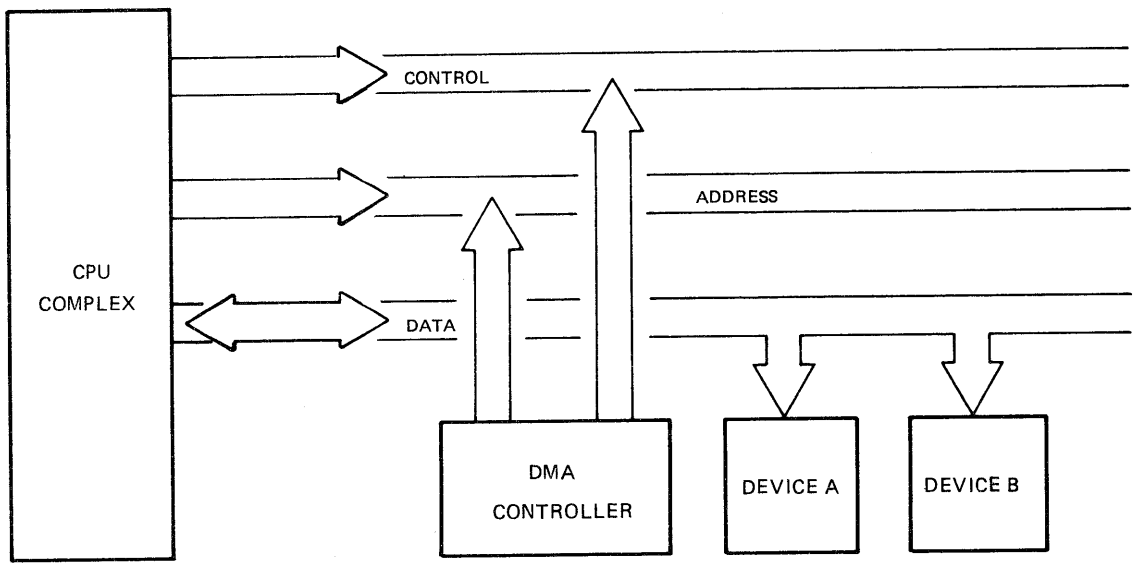


Fig. 2-6 Simple DMA Transfer

DMA CONTROLLER FUNCTIONAL DESCRIPTION

The DMA Controller used on the system is the Intel 8237. Figure 2-7 is a block diagram of the DMA Controller (DMAC).

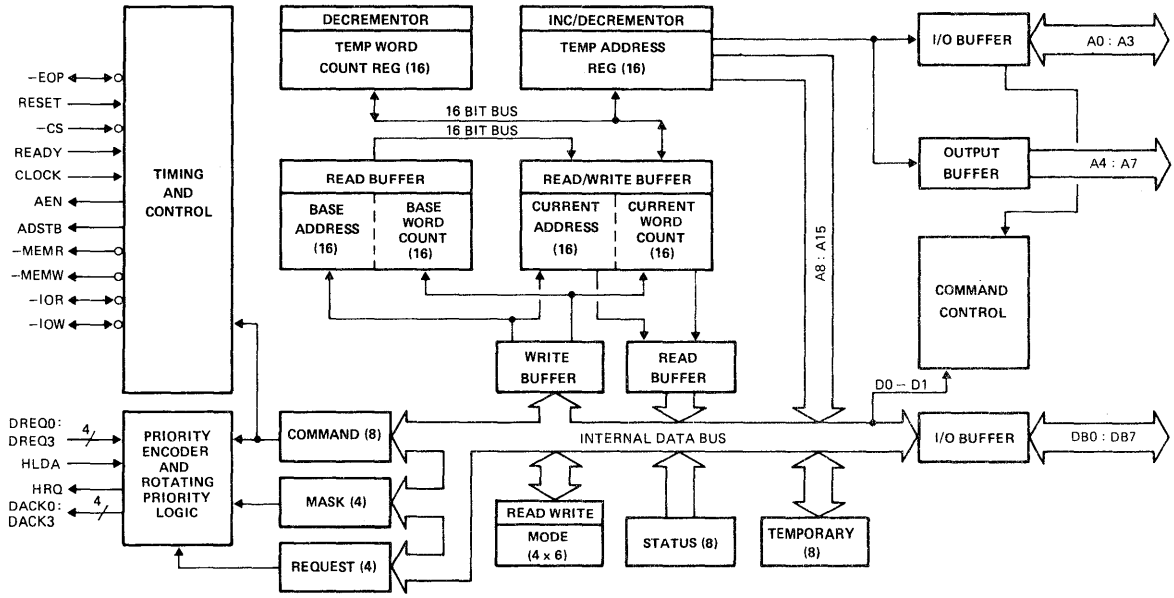


Fig. 2-7 DMA Controller (8237) Block Diagram

The DMA Controller (DMAC) contains three basic blocks of control logic.

The Timing Control Block that generates internal timing and external control signals for the DMAC.

The Program Command Control Block that decodes the various commands given to the DMAC by the microprocessor prior to servicing a DMA request. It also decodes the Mode Control word used to select the type of DMA during the servicing.

The Priority Encoder Block that resolves priority contention between DMA channels requesting service simultaneously. Channel 0 has the highest priority and channel 3 has the lowest priority.

Channel 0 is used to refresh the system dynamic memory. A channel of the timer is programmed to request periodically a dummy DMA transfer. This creates a memory-read cycle, which is available to refresh the dynamic memory on the motherboard as well as the memory expansion boards.

Channel 1 is connected to the I/O expansion bus to support high speed data transfer between I/O devices and memory.

Channel 2 is dedicated to the mini-floppy disk controller for transferring information to/from the minifloppy disk drive.

Channel 3 is connected to the I/O expansion bus to support high speed data transfer between I/O devices and memory.

DMA OPERATION

The DMA is designed to operate in two major cycles, the idle cycle and the active cycle.

Each device cycle is made up of a number of states. The DMAC can assume seven separate states, each composed of one full clock period. State I (SI) is the inactive state. It is entered when the DMAC has no valid DMA requests pending. While in SI the DMAC is inactive but may be in the Program Condition, being programmed by the processor.

State 0 (S0) is the first state of the DMA service, when the DMAC has requested a hold but the processor has not yet returned an acknowledge. The DMAC may still be programmed until it receives HLDA (hold acknowledge) from the CPU. An acknowledge from the CPU indicates that DMA transfers may begin.

S1, S2, S3 and S4 are the working states of the DMA service. If more time is needed to complete a transfer than is available with normal timing, one WAIT cycle must be inserted. This is automatically inserted by part of the programmable WAIT logic

The WAIT state is inserted between S2 or S3 and S4 by the use of the READY line (DMARDY) on the DMAC. Note that the data is transferred directly from the I/O device to memory (or vice versa) with -IOR and -MEMW (or -MEMR and -IOW) being active at the same time. The data is not read into or driven out of the DMAC in I/O to memory or memory to I/O DMA transfers.

DMA CIRCUITRY

The DMAC issues a DMA Hold Request Acknowledge DMAHRQA whenever there is at least one valid DMA request. DMAHRQA goes to an LS244 latch pin 8 and is output as DMAHRQ. DMAHRQ then inputs a 74S174 D latch which is clocked by the signal -CLK8. The output SDMAHRQ is one of the inputs to the Programmable Array Logic Chip PAL16R8. This PAL forms part of the Arbiter Logic explained in section 2-8.

Figure 2-8 shows the DMA circuitry used on the system.

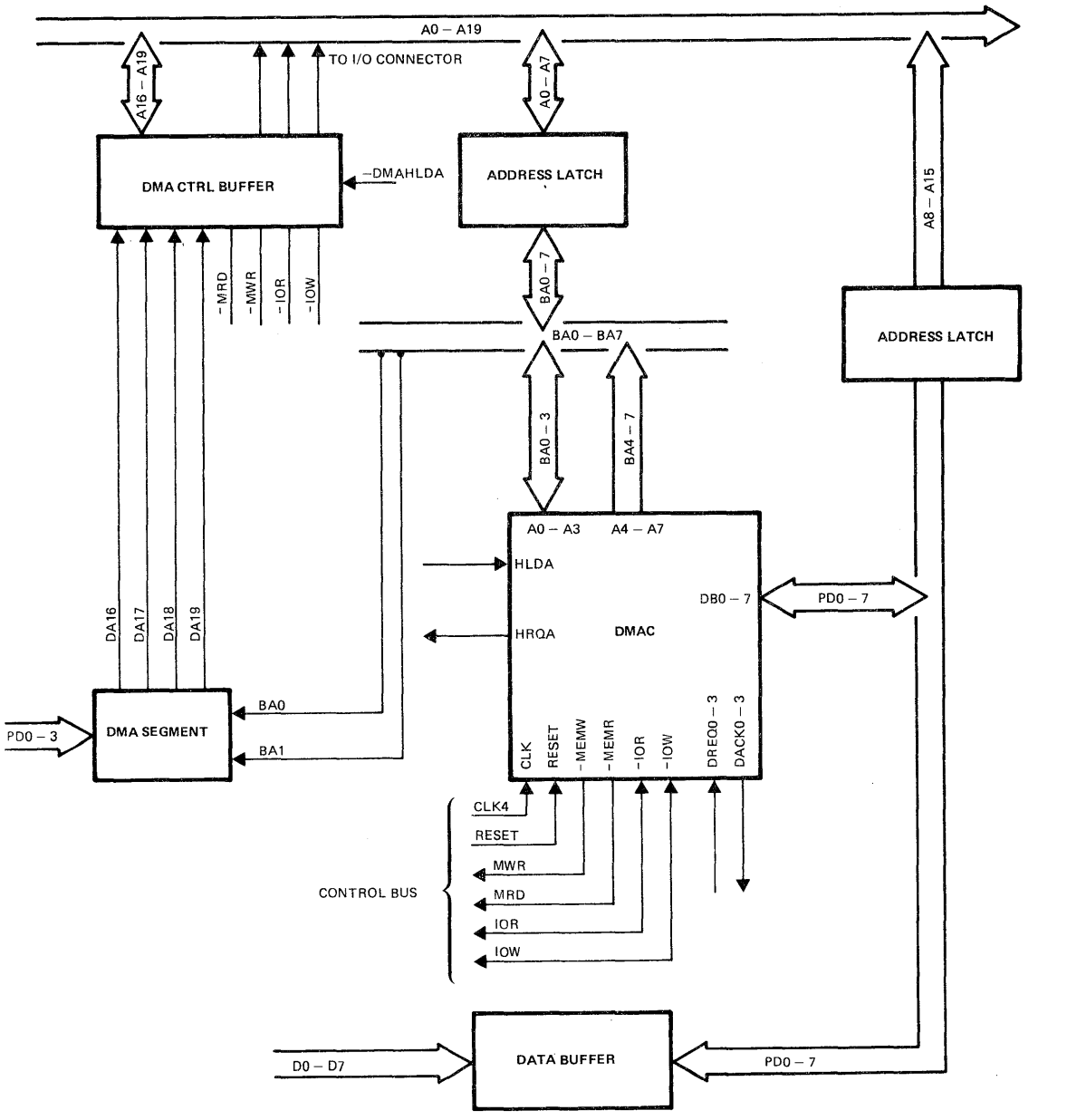


Fig. 2-8 DMA Circuitry

The PAL outputs -DMAHLDA (DMA hold acknowledge) and -P86GT become active. The -DMAHLDA does a number of things:

- It passes through an inverter (LS04) to become DMAHLDA and then goes to the AND gate LS08 where it is ANDed with a normally high signal (pull up resistor present on other input) to produce the signal XHA. This signal gives the DMAC the go-ahead to take control of the address bus, data bus and control bus. Also present on the AND input is -XHLDMA, which is the Hold Acknowledge signal for devices connected to the I/O connectors.

- It is input to pin 1 of the DMA Control Buffer and its effect is to change the direction of data flow through the buffer from B to A. This enables the DMAC to dialogue with devices such as expansion memory boards and I/O boards not on the motherboard.

- It enables the DMA address latch and the BA bus buffer to permit the DMA Controller to transfer bytes from the BA and DMA busses onto the A bus.

The generation of the signal -P86GT from PAL16R8 has the effect of disabling the CPU Address Latches and the CPU Data Buffers and the command lines from the Bus Controller. Hence the Address, Data and Control Busses are isolated from the CPU.

The DMAC now assumes control of the system busses.

The address for the first transfer operation comes out in two bytes. The least significant 8 bits (BA0-BA7) are output onto the the address bus via the BA bus buffer while the most significant 8 bits (PD0-PD7) are first strobed into the DMA address latch by the signal ADSTB before being output onto the address bus to complete the full 16 bits of the address bus. After the initial transfer takes place , the latch is updated only after a carry or borrow is generated in the least significant address byte.

DMAC supports 16 bit addresses which allow addressability to 64KB of memory. In order to make it able to address 1MB of memory as the CPU does, a 4 x 4 register file is used to provide the upper 4 bits DA16-DA19. This register file is named DMA segment register and is programmable by software to address any 64KB block within 1MB of memory space.

On the following page is a timing diagram showing the sequence of some of the signals used during a DMA operation.

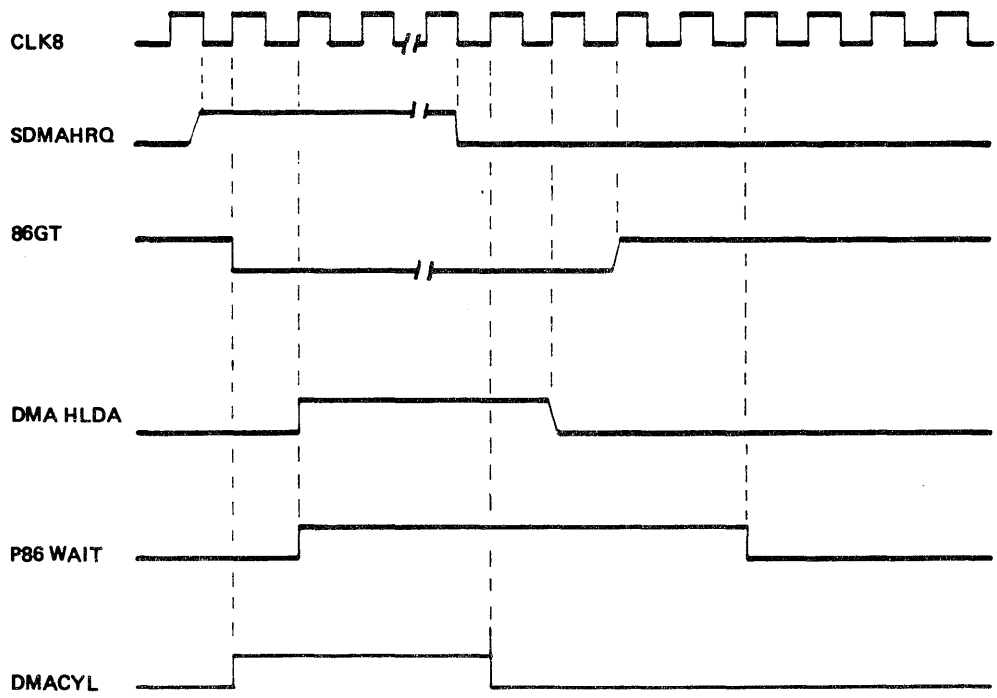


Fig. 2-9 Timing Diagram

DMA - DEVICE SERVICE

Idle Cycle

When no channel is requesting service, the DMAC will enter the idle cycle and perform SI states. In this cycle the DREQ lines are sampled every clock cycle to determine if any channel is requesting a DMA service. When -CS is low and HLDA is low, the DMAC enters the program condition and address lines A0 to A3 address the control register to be loaded or read. The -IOR and -IOW lines are used to select and to time reads or writes.

Active Cycle

When the DMAC is in the Idle Cycle and a channel requests a DMA service, the DMAC will output an HRQ and enter the active cycle. It is in this cycle that the DMA service will take place, in one of four modes:

Single Transfer Mode

In single transfer mode the device is programmed to make one transfer only. The word count will be decremented and the address decremented or incremented following each transfer. When the word count "rolls over" from zero to FFFF hex, a Terminal Count (TC) will cause an Autoinitialize if the channel has been programmed to do so.

DREQ must be held active until DACK becomes active in order to be recognized. If DREQ is held active throughout the single transfer, HRQ will go inactive and release the bus to the system. It will go active again and, upon receipt of a new HLDA, another single transfer will be performed.

Block Transfer Mode

In Block transfer Mode the device is activated by DREQ to continue making transfers during the service until a TC, caused by word count going to FFFF hex or an external -EOP (End of Process), is encountered. DREQ need only be held active until DACK becomes active. Again, an Autoinitialization will occur at the end of the service if the channel has been programmed for it.

Demand Transfer Mode

In Demand Transfer mode the device is programmed to continue making transfers until a TC or external -EOP is encountered or until DREQ goes inactive. Thus transfers may continue until the I/O device has exhausted its data capacity. After the I/O device has a chance to catch up, the DMA service is re-established by means of DREQ. During the time between services when the CPU is allowed to operate, the intermediate values of address and word count are stored in the DMAC Current Address and Current Word Count registers. Only an -EOP can cause an Autoinitialize at the end of the service. -EOP is generated either by TC or by an external signal.

INTERRUPT CONTROL LOGIC

Hardware interrupts are asynchronous events requiring CPU attention and are generally initiated by peripheral devices requiring service. The CPU is normally allowed to execute its main program until an interrupt request occurs. On receipt of an interrupt request, the CPU completes the instruction being executed, saves its current state (instruction pointer, code segment and flags) and fetches a new routine to service the interrupting device. Once the interrupting device has been serviced, the CPU resumes its main program at the point where it was interrupted.

The system has 9 interrupts in all: eight maskable priority interrupts and one non-maskable interrupt. The descending order of priority for these interrupts and associated devices are listed below.

NMI	non-maskable interrupt
IRQ0	generated by Timer channel 0
IRQ1	generated by the Keyboard Controller
IRQ2	generated by a board on one of the expansion slots
IRQ3	generated by a board on one of the expansion slots
IRQ4	generated by the Serial Controller
IRQ5	generated by a board on one of the expansion slots
IRQ6	generated by the Floppy Disk Controller
IRQ7	generated by the Parallel Interface

NMI is handled by the NMI logic circuitry while IRQ0-7 are handled by the Interrupt Controller chip.

NON-MASKABLE INTERRUPT

The NMI has the highest priority. It is a hardware generated interrupt and is so called because the interrupt cannot be turned off by clearing the interrupt flag. The simplified NMI circuitry is shown in figure 2-10.

In this system an NMI is generated by any of the following events:

1. Memory Parity Error (signal -MBMERR causes train of events to activate NMI)

2. 8087 NDP Interrupt (signals 87INT and SW004 cause train of events to activate NMI)

3. Failure in power-up diagnostics (signals -CHCK and -IOCHK cause train of events to activate NMI)

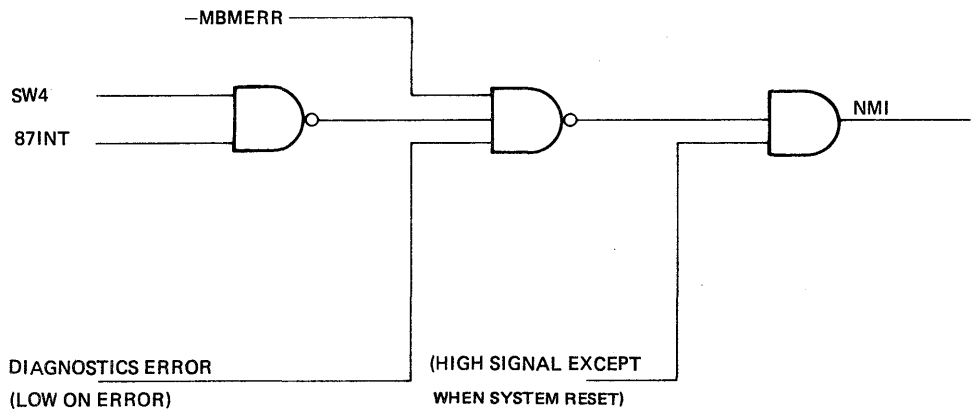


Fig. 2-10 NMI Logic

The NMI is edge triggered into the CPU on a low to high transition. NMI is not synchronized to the clock but must have a duration of more than two CLK cycles in the high state. Any high going transition of NMI is latched on the 8086 CPU chip and is serviced at the end of the current instruction or between whole moves of a block-type instruction. The low going edge of NMI may occur before, during, or after the servicing of NMI. Another high going edge triggers another response if it occurs after the start of the NMI procedure.

MASKABLE PRIORITY INTERRUPTS

The circuitry that handles the maskable priority interrupts (IRQ00 - IRQ70) is the Intel 8259 Interrupt Controller chip. This Controller is a programmable device which functions as an overall manager in the interrupt-driven system environment. In other words, this controller has the task of only letting one device through to the CPU at a time, taking into consideration the fact that some devices have a higher priority than others. Fig. 2-11 is a block diagram of the 8259 Interrupt Controller.

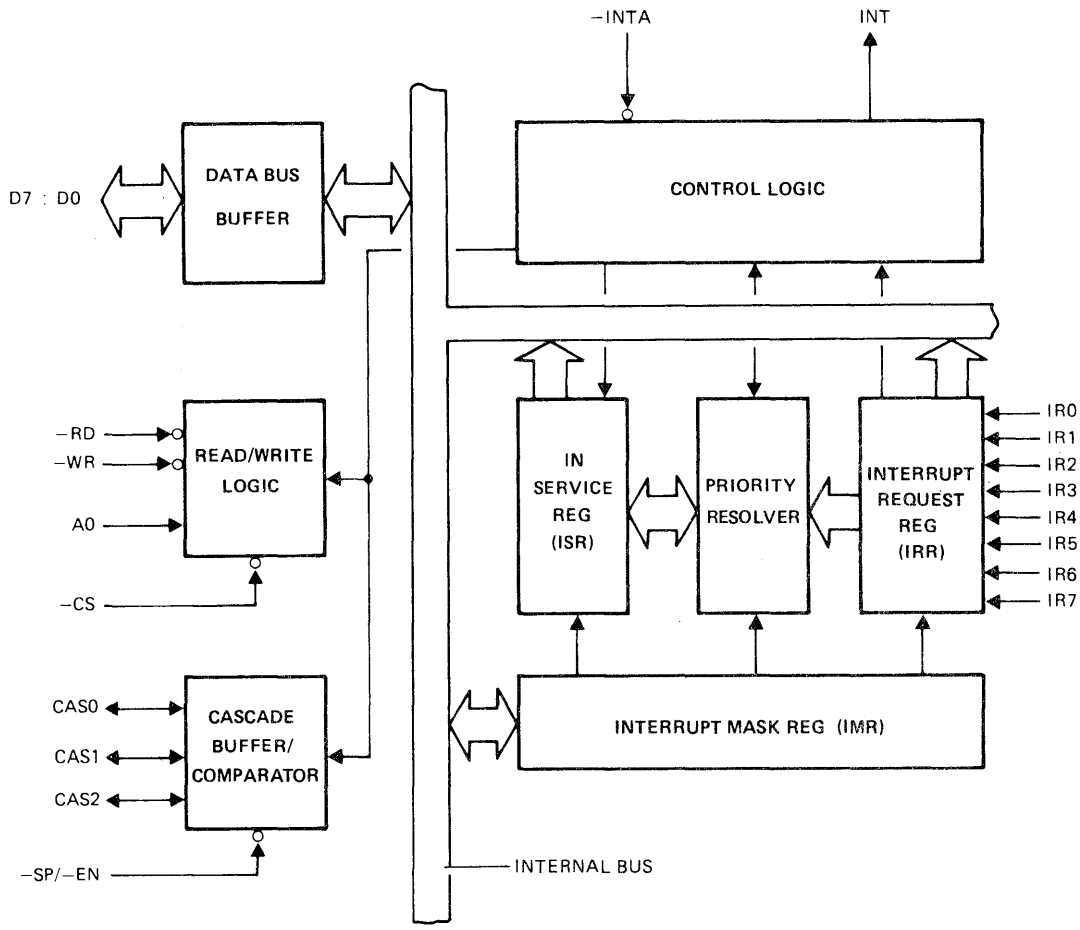


Fig. 2-11 Block Diagram of Interrupt Controller

Each device has an interrupt line which runs to one of the eight interrupt lines of the Programmable Interrupt Controller (PIC). The 8259 PIC can be programmed to ignore or to monitor any combination of these lines. This selection is determined via what is called the interrupt mask, a byte which is sent to the PIC by the CPU through a port located in the I/O space. This port is known as the control port. The 8 bits in this mask correspond to the eight devices. To turn off interrupts from a device, one has merely to set the corresponding bit in the mask equal to 1. Thus the 8259 will ignore all eight devices if a mask of FF hex is sent, and will respond to all of them if a mask of 00 hex is sent.

If two or more devices (which are not masked out) signal the PIC for service at the same time, the PIC determines which goes first according to several user-selectable schemes. These include a fixed priority and a rotating priority scheme. The devices not yet serviced wait their turn in a waiting area which the PIC keeps track of. When a device begins to be serviced, the PIC moves out of this waiting area into the working area.

There is one interrupt line which runs from the PIC to the interrupt input line (INTR0) on the 8086 CPU. When the CPU gets a request on this line, it sends an acknowledge signal on the interrupt acknowledge line (INTA). The 8259 then sends a byte to the CPU telling it what type (location in the interrupt table) of interrupt should be executed. The value of the interrupt type is programmable. The eight devices are assigned eight consecutively numbered interrupt types starting at any multiple of 8.

CLOCK GENERATOR

The system uses the Intel 8284A as the Clock Generator chip. This is used in conjunction with a 24MHz crystal oscillator and a LS241 buffer to supply the timing for the entire system.

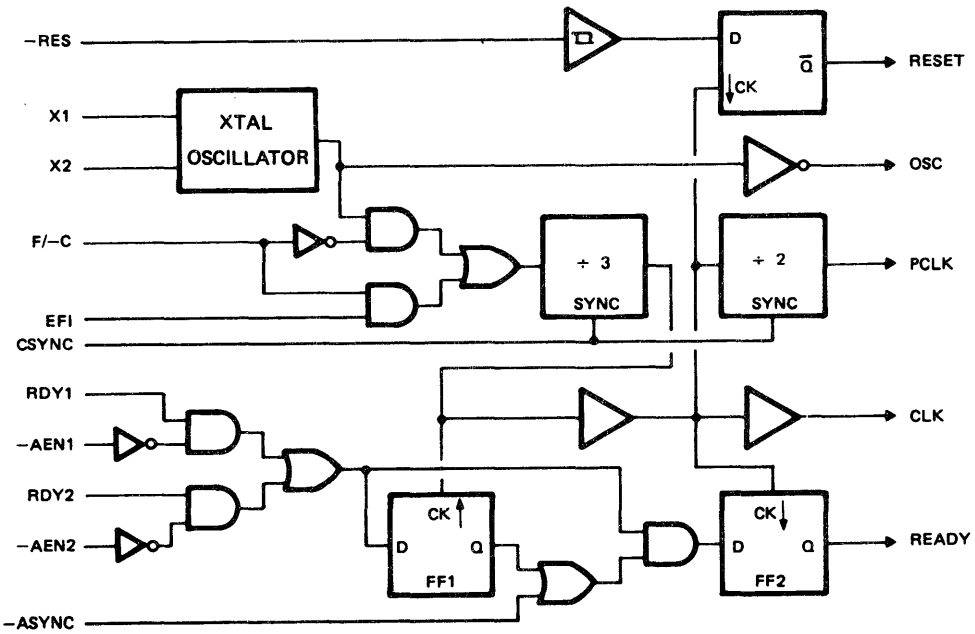


Fig. 2-12 8284 Clock Generator Block Diagram

The OSC output is a TTL level output of the internal oscillator circuitry. Its frequency is equal to that of the crystal. OSC is made available for use on the display controller board.

The clock signal CLK generated by the 8284 is an 8MHz clock with a 33% duty cycle and is used by the 8086 CPU. The 8284 also generates a 50% duty cycle 4MHz clock, PCLK, which is used by peripherals. In order to provide enough driving capability and to reduce the clock skew, these clocks are buffered and repowered by LS241.

The following are the outputs of LS241.

CLK86 This is an 8MHz clock and used mainly by the 8086 CPU, 8087 NDP and 8288 bus controller.

BCLK8 An 8MHz clock which travels on the C bus and is used by the Bus Arbiter, Programmable Wait Logic and I/O Chip and Port Select Logic.

XCLK8 An 8MHz clock which is made available on the I/O expansion bus.

CLK8 An 8MHz clock which is used by the Bus Arbiter, Progammable Wait Logic and Memory Control Logic.

CLK4 A 4MHz clock used mainly by the DMA controller, Floppy Disk controller and keyboard controller.

XCLK4 A 4MHz clock which is made available on the I/O expansion bus.

-RESET1 This signal is used to reset various parts of the logic on the motherboard whenever the reset switch is pressed.

XRESET This is made available as a reset signal to the I/O expansion bus.

There are two signals which go to/from the 8284 Clock Generator to the 8086 processor and 8087 NDP. These are the RESET and READY signals. They are routed through the clock generator to synchronize them with the clock signals.

The function of the RESET signal is to restart the processor as if it had been turned off and on again. This is done in situations such as an infinite loop or a power glitch which has affected parts of the program in memory.

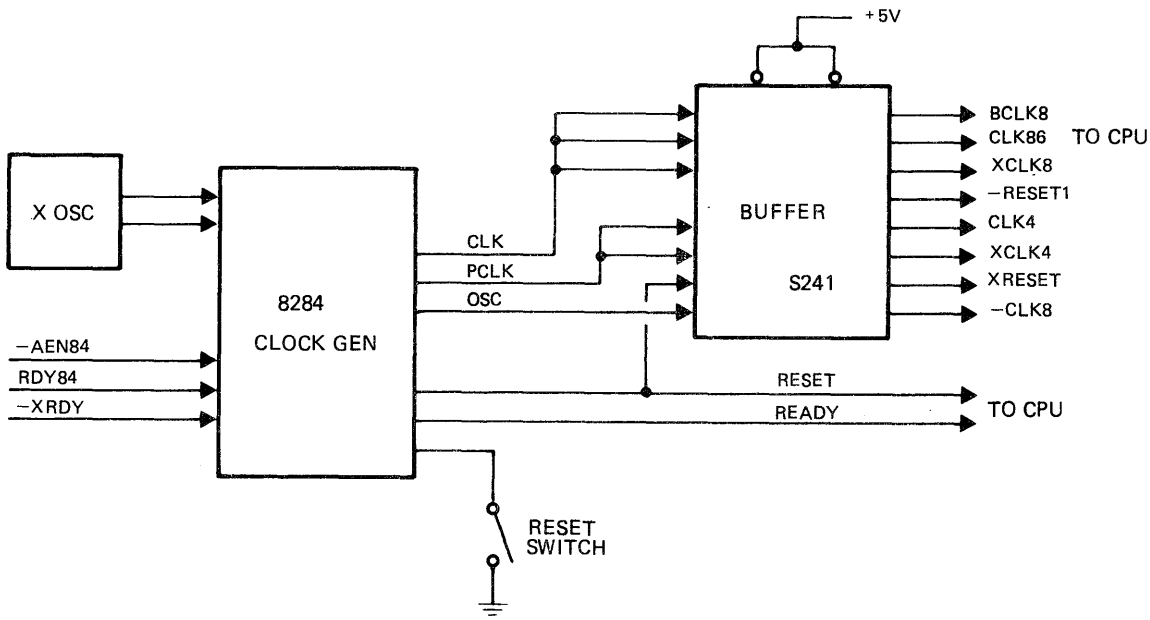


Fig. 2-13 Clock Generator

The function of the READY signal is to insert the required number of "wait" states for slow memory or I/O devices. The READY signal goes from the external device through the clock generator to the processor. When the processor requests access to a device which is not ready to make a transfer, the device sends a low over the ready line and stops the processor until the READY signal goes high again. This is done by controlling the inputs to the clock generator by the signals -AEN84 and RDY84.

The signal -AEN84 is normally low but goes high whenever "wait" states are to be inserted. The signal RDY84 goes low when either the DMA controller or an external processor require the CPU to stop.

Thus the READY output of the 8284 goes low when either the signal RDY84 is low (meaning that the DMA controller or the external processor are in command of the bus) or else the signal -AEN84 is high (meaning that one or more "wait" states need to be inserted).

The ready signal remains low until RDY84 and -AEN84 both go active. When both are active then READY goes high and the 8086 CPU starts functioning again.

BUS ARBITER

The bus arbiter arbitrates between the three system bus masters. These masters are:

- 8086 CPU

- 8237 DMA Controller

- External Processor, Concurrent or Alternate

The bus arbitration is handled by the PAL16R8 which is used as a sequencer to sample the bus requests from the DMA controller and the external processor during the time that the CPU status is passive and not in the lock state. It then arbitrates the system bus to the DMA controller or an external processor, the DMA controller always taking priority over an external processor.

Once a bus request has been made by one of the above bus masters and is accepted by the bus arbiter, the appropriate address latches and data buffers are enabled and/or disabled, the CPU is put into the wait state and the new bus master acknowledged. The new bus master then takes control of the system busses.

When the new bus master has finished with the system bus, it removes its bus request and the bus arbiter gives control back to the CPU. The whole arbitration cycle is then repeated.

If the external processor is an alternate processor, once it has control of the system bus the CPU will not regain control until a hardware reset is performed. In this case the bus arbiter only arbitrates between the external processor and the DMA controller.

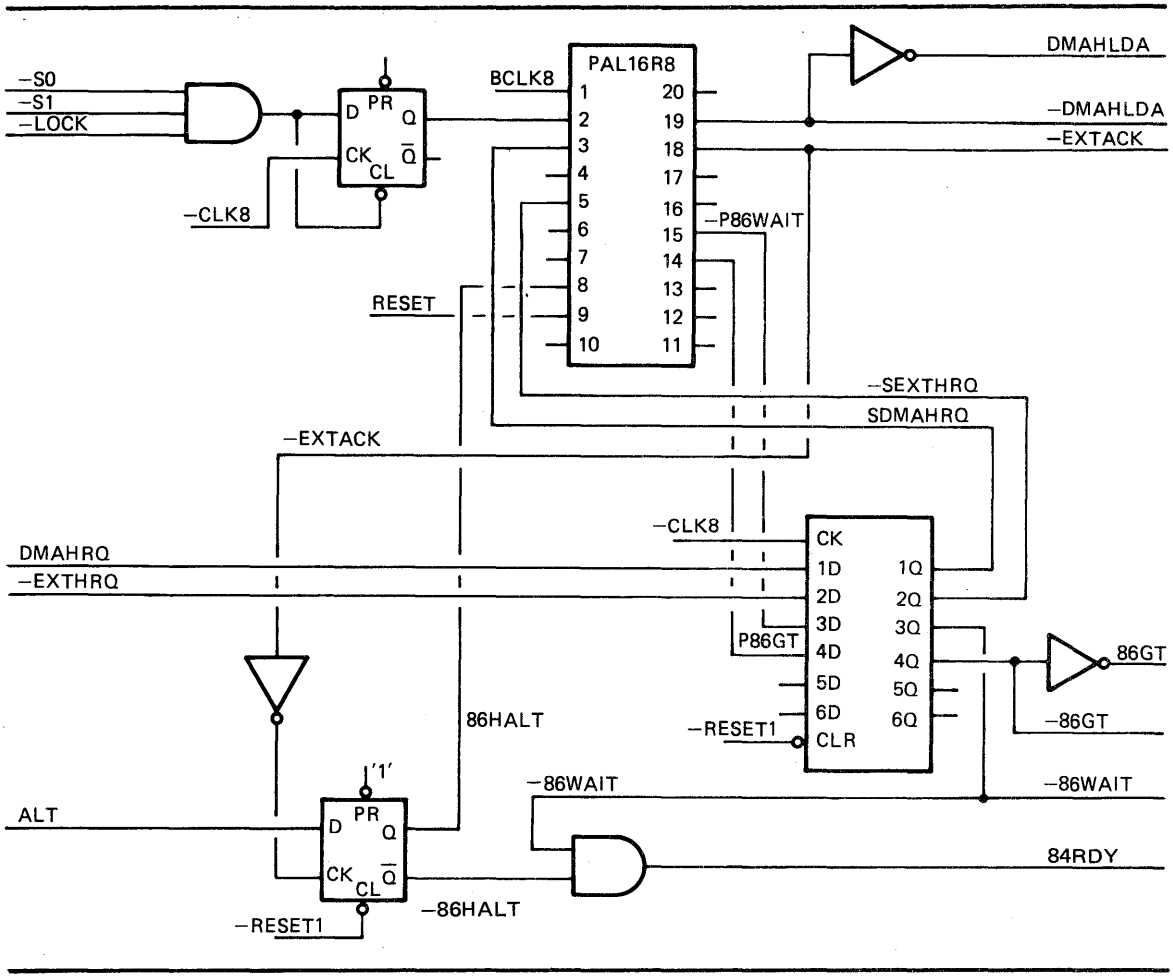


Fig. 2-14 Bus Arbiter

DMA CONTROLLER CYCLE

The sequencer samples the DMA controller and external processor request lines DMAHRQ and EXTHRQ via the bus arbiter latch. If a DMA Controller request is present, irrespective of an external processor request, the bus arbiter issues the -P86GT signal. After passing through the bus arbiter latch, this signal changes to -86GT and is used to disable the CPU address latch, bus controller and data bus buffer. Thus the CPU address/data bus is isolated from the system address and data busses. The sequencer then starts the DMA controller cycle that issues the -P86WAIT signal together with -DMAHLDA and DMAHLDA.

The -P86WAIT signal becomes -86WAIT and is AND gated with -86HL1 to produce the RDY84 signal that acts on the ready input of the 8284 clock generator to make the CPU ready signal inactive. This puts the CPU into the wait state.

The -DMAHLDA signal changes the direction of transmission through the DMA control buffer, so that the expansion memory read and write signals and the address bits A16 to A19 are derived from the DMA controller. It

enables the DMA segment register and the DMA controller high byte address buffer LS373, and also changes the direction of the DMA controller low byte address latch so that the address bits A0-A16, which go on the A bus, are derived from the DMA controller.

The DMAHLDA signal indicates to the DMA controller that the CPU has released control of the system busses.

When the DMA Controller has finished with the system bus the DMAHRQ is removed. The sequencer then removes the DMAHLD signal followed by -P86GT and -P86WAIT so that control of the system bus is given back to the CPU.

EXTERNAL CYCLE, CONCURRENT AND ALTERNATE

If an EXTHRQ is present, without a DMAHRQ being present, the bus arbiter issues the -P86GT signal to isolate the CPU address/data bus from the system address and data busses. The sequencer then starts the external processor cycle that issues the -P86WAIT signal together with EXTACK.

The -P86WAIT signal initiates a CPU wait state in the same manner as for the DMA controller cycle, and the -EXTACK signal clocks the ALT bit into the external processor latch to set the state of the 86HALT signal. The external processor is now able to take control of the system bus.

If the external processor is to act as a concurrent processor, the ALT bit will have been set low. This low, clocked into the external processor latch by the EXTACK signal, will maintain the 86HALT signal low and the sequencer treats the external processor as a concurrent processor.

When the external processor has finished with the system bus, the EXTHRQ is removed. The sequencer then removes the EXTACK signal followed by -P86GT and P86WAIT so that control of the system bus is given back to the CPU.

If the external processor is to act as an alternate processor, the ALT bit will have been set high. This high, clocked into the external processor latch by the EXTACK signal, sets the 86HALT signal high and the sequencer branches to the alternate processor cycle. The -86HALT signal, now set low, takes the RDY84 signal low and the CPU is put into the wait state. This maintains the ALT bit high and the CPU is held in the wait state until a hardware reset is performed.

In this case the bus arbiter only arbitrates between the external processor and the DMA controller.

BUS CONTROLLER

The motherboard uses an Intel 8288 Bus Controller to perform the control of the various busses. This controller decodes the status output of the CPU in order to generate the bus command and control signals at the appropriate times. A block diagram of the Bus Controller is shown in Figure 2-15.

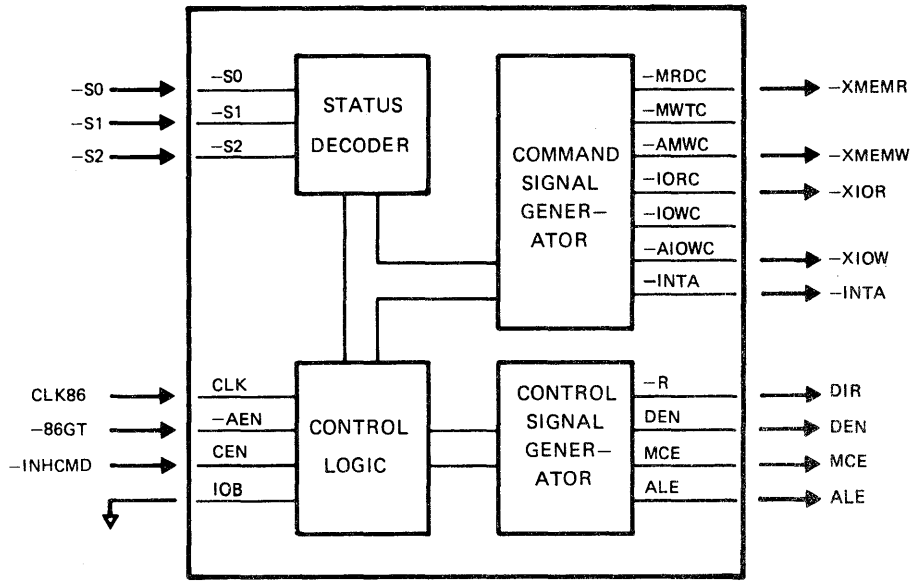


Fig. 2-15 8288 Bus Controller Block Diagram

The command signals issued are as follows:

Status Bits S2	S1	S0	Signal	8288 Command Signals
0	0	0	-INTA	Interrupt Acknowledge
0	0	1	-XIOR	Read I/O
0	1	0	-XIOW	Write I/O
0	1	1		none
1	0	0	-XMEMR	Read Memory
1	0	1	-XMEMR	Read Memory
1	1	0	-XMEMW	Write Memory
1	1	1		none

The control signals issued are as follows:

8288 Control Signals	Function
Address Enable (ALE)	to strobe the address onto the address latches
Data Enable (DEN)	to enable the data buffers onto the data busses
Direction (DIR)	to establish the direction of data flow through the data buffers

The Bus Controller outputs are enabled when the enable signal -86GT becomes active. Commands are only issued 115ns after this signal becomes active so that memory and I/O commands wait for bus arbitration. -86GT going inactive immediately tri-states the command outputs.

PROGRAMMABLE WAIT LOGIC

The programmable wait logic is used to insert the correct number of wait cycles required by the various memory and I/O circuits. It primarily consists of a PROM, that is essentially a look-up table for the number of wait cycles required for the various I/O circuits, and a divide-by-16 counter. Figure 2-16 shows the logic diagram of the programmable wait logic.

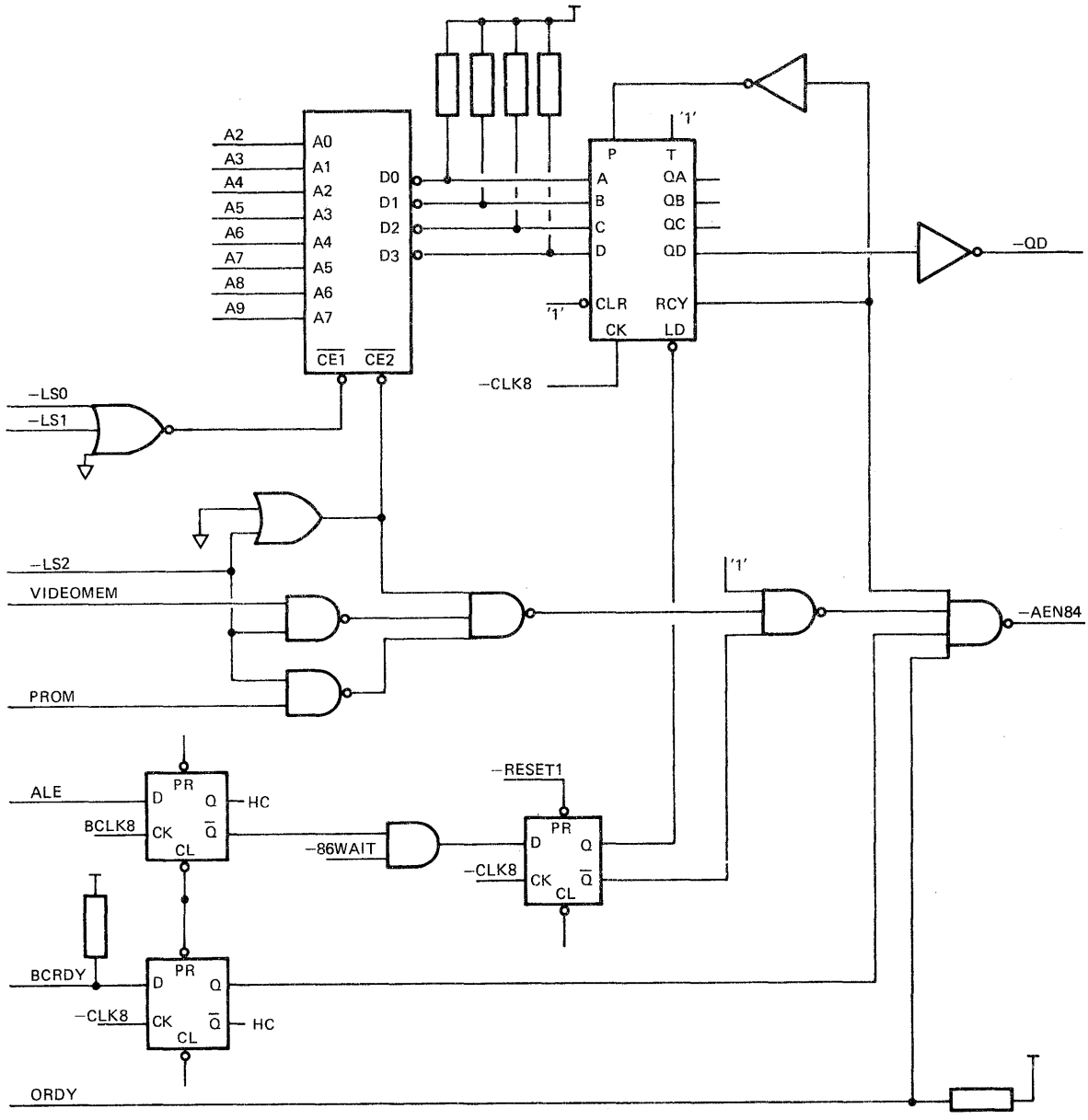


Fig. 2-16 Programmable Wait Logic

The PROM input consists of the I/O addresses A2 to A9. Its output, loaded into the counter by the ALE signal from the bus controller or by the -86WAIT signal from the bus arbiter, is 1's compliment of the number of wait cycles required.

Whenever the counter is loaded with F hex or the counter counts up to F hex, it generates a carry signal -WRDY. This carry signal is input to the clock generator enable NAND gate together with the ORDY and BCRDY signals from the expansion boards and the composit VIDEOMEM/PROM wait signal in order to produce the signal -AEN84. This signal acts on the address enable input -AEN1 of the 8284 Clock Generator to make the CPU ready signal active and maintain the CPU in the ready state. The carry output is fed back to the P enable input of the counter in order to inhibit the counter at F hex and maintain the CPU ready.

Loading the counter with a value other than F Hex, 1's compliment of number of wait cycles required, removes the carry signal and makes the CPU ready input inactive so that the CPU enters the wait state. The CPU then waits until the counter counts to F hex. In this manner the required number of wait cycles are inserted.

For memory access the wait logic PROM is disabled, the memory read and write statuses being decoded from the CPU status lines to produce the wait logic PROM enable signal. In order to insert wait cycles required for video memory access and system PROM access, signals VIDEOMEM and PROM are NAND gated with the status line -LS200 and the resultant signals are further NAND gated with the wait logic PROM enable signal in order to produce the composit VIDEO/PROM wait signal. This is NAND gated with the -Q output of the counter load latch and input to the clock generator enable gate to produce signal -AEN84.

For the DMA cycle, one wait cycle is always inserted by the DMA wait logic. This wait logic acts on the DMA Controller ready input to extend the memory read and write pulses. The timing of the DMA ready signal is determined by the DMA address strobe ADSTB and the 4 MHz clock CLK4.

RANDOM ACCESS MEMORY

The motherboard RAM memory is organized in such a way that it could be addressed as bytes as well as words (2 bytes). It is divided in two banks namely bank 0 and bank 1 and each bank has an even byte bank and an odd byte bank. The motherboard has a minimum of 128K x 9 of read/write memory and is expandable up to a maximum of 640K x 9 of RAM with chip insertion.

MEMORY CONTROL LOGIC

As shown in figure 2-17 the signals required to address and access memory are:

-RASL	Row Address Strobe Low
-RASH	Row Address Strobe High
-CAS0	Column Address Strobe for bank 0
-CAS1	Column Address Strobe for bank 1
MA0-MA8	Memory Address Lines
MD0-MD15	Memory Data Lines
-MWR	Write Enable
PALIN	Parity Low Input
PAHIN	Parity High Input
PALOUT	Parity Low Output
PAHOUT	Parity High Output

The memory cycle starts from the decoding of the CPU status lines -S0 to -S2. When these status lines are for memory read, memory write or code access, the input to the flipflop LS112 goes low. Thus when the flipflop is clocked the output signal LDRAM goes active high.

For the generation of -CAS0 and -CAS1 signals the LDRAM signal is delayed by the delay line PE21388 to allow the row address to be latched into the RAM memory.
For memory read or code access, -CAS0 and -CAS1 are delayed by 80ns while for a memory write -CAS0 and -CAS1 are delayed by 160ns to give time for the data to be latched into the memory data buffers.

PAL14H4 decodes whether the memory address is in bank 0, bank 1, in video memory or in the PROM. This is done from the system memory configuration switch SW0-SW3 and the address bits A13-A19.

If the memory address is for motherboard RAM memory the PAL generates either BANK0 or BANK1 to select the corresponding column address strobe -CAS0 or -CAS1.

The selection of -RASL or -RASH is done by the signals A0 and -BHE as seen in the table below.

-BHE	A0	-RASL	-RASH	
0	0	0	0	Word (2 bytes)
0	1	1	0	Upper byte from/to odd address
1	0	0	1	Lower byte from/to even address
1	1	1	1	NONE

This means that on a word operation both -BHE and A0 are low and both -RASL and -RASH are activated. On a byte operation either -BHE is high for an even address selecting -RASL or A0 is high for an odd address activating -RASH.

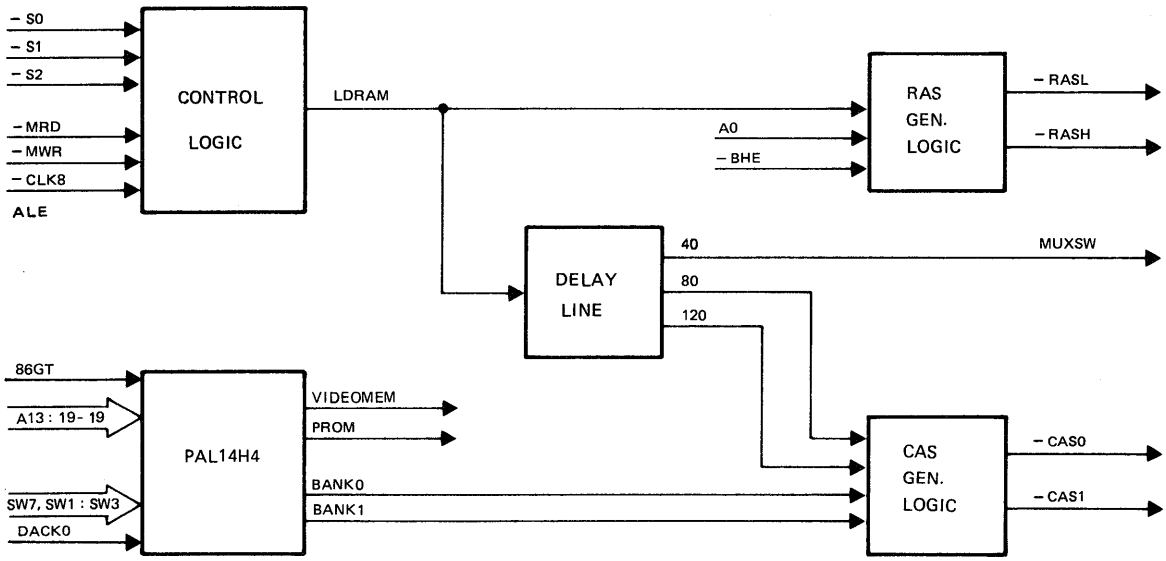


Fig. 2-17 Memory Control Circuitry

MEMORY ADDRESSING

The addresses for the DRAM memory travel on the A bus. A1 to A8 go on to the RAM address multiplexer which consists of two 2 to 1 74S150 multiplexers and four AND gates.

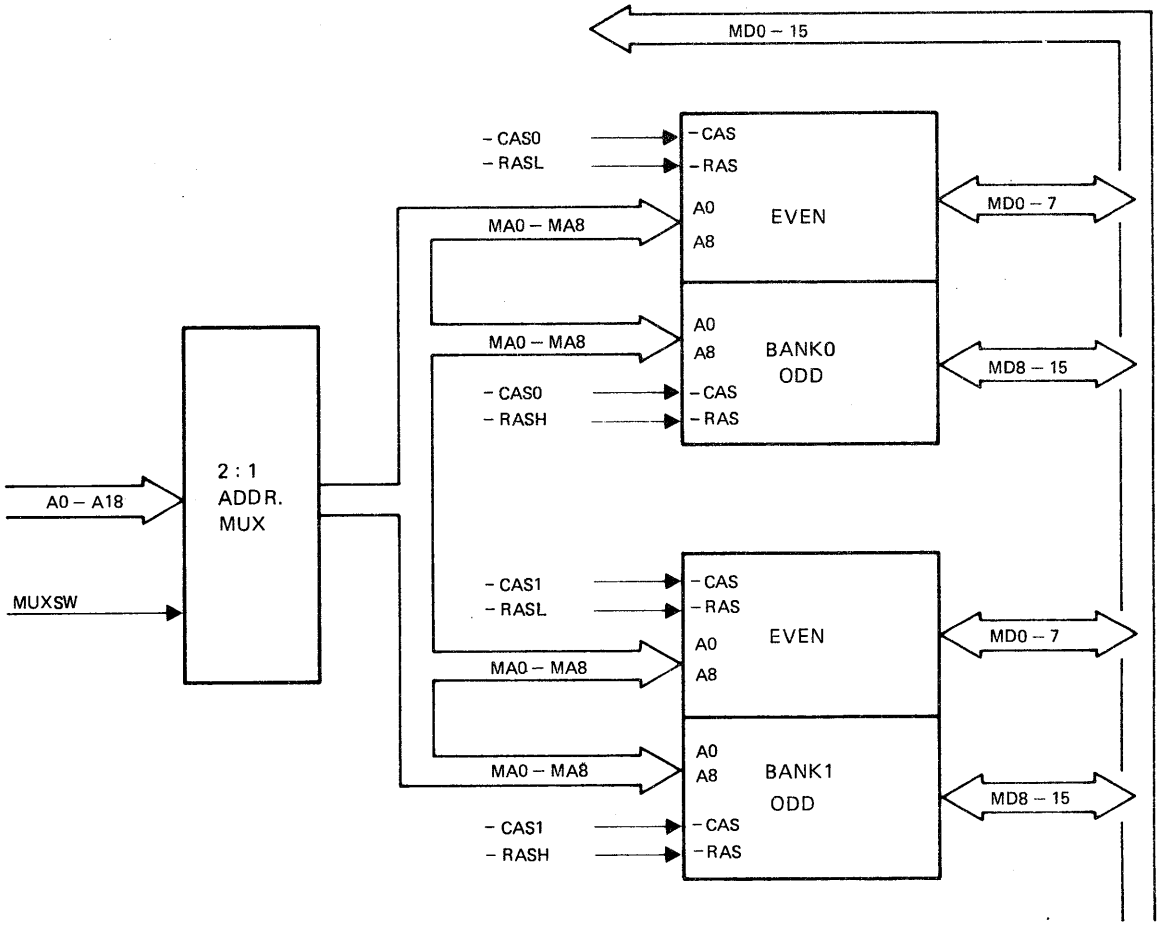


Fig. 2-18 Memory Addressing Logic

When the address bits arrive at the multiplexers, the select input pin 1 signal MUXSW is low and so the nine bits A1-A8,A17 go on to the output of the multiplexers. Thus when the -RASL or -RASH or both signals go active the outputs MA0-MA8 of the multiplexers are latched into the RAMs as the row address for the odd or even or both bytes (in the case of word addressing) for the particular bank.

40ns after -RASL or -RASH or both go active, MUXSW signal goes high, and so the address bits A9-A16, A18 go to the output of the multiplexers. Thus when the -CAS0 or -CAS1 signals go active, MA0-MA8 are latched into the RAMs as the column address for the even and odd bytes for either bank 0 or bank 1.
In a read cycle -CAS0 or -CAS1 become active 80ns after -RASL or -RASH and 160ns in a write cycle.

Data from the addressed DRAMs passes through the 16 bit memory data buffer which consists of two bidirectional buffers (74LS245). This data buffer connect the MD bus to the D bus. Odd addressed data goes on MD8-MD15 while even addressed data goes on MD0-MD7.

Therefore on a word operation (16 bits to be accessed) two possibilities exist:

- First byte on even address
- First byte on odd address

If the first byte is on an even address only one memory cycle is required for accessing two bytes of data since both are on the same bank. But if the first byte being addressed is on an odd address, two cycles are required for word operation since the two bytes are on different banks.

Data is input or output from/to the MD bus according to the state of the write enable signal -MWR at the RAMs. If -MWR is low then a write operation occurs and data from the MD bus is input to the addressed RAMs. If -MWR is high then a read operation occurs and data from the addressed RAMs is output on the MD bus.

On a DMA memory cycle one wait state is inserted. This is generated by the circuitry consisting of the two D flipflops and an AND gate 74LS11. The address strobe ADSTB signal from the DMA controller, besides latching the upper address byte, is also input to the DMA wait logic. On a low to high transition of this signal the output Q of the first F/F, the output of the AND gate, and the output Q of the second F/F all go low. The latter Q output, called DMARDY, is used as an input to the DMA controller. Its function is to extend the memory read and write signals from the DMA controller.

DMARDY, as stated above, goes low and forces the DMAC to add wait states till the DMARDY goes high again. The timing in the system requires only one WAIT state to be inserted during a DMA memory cycle.

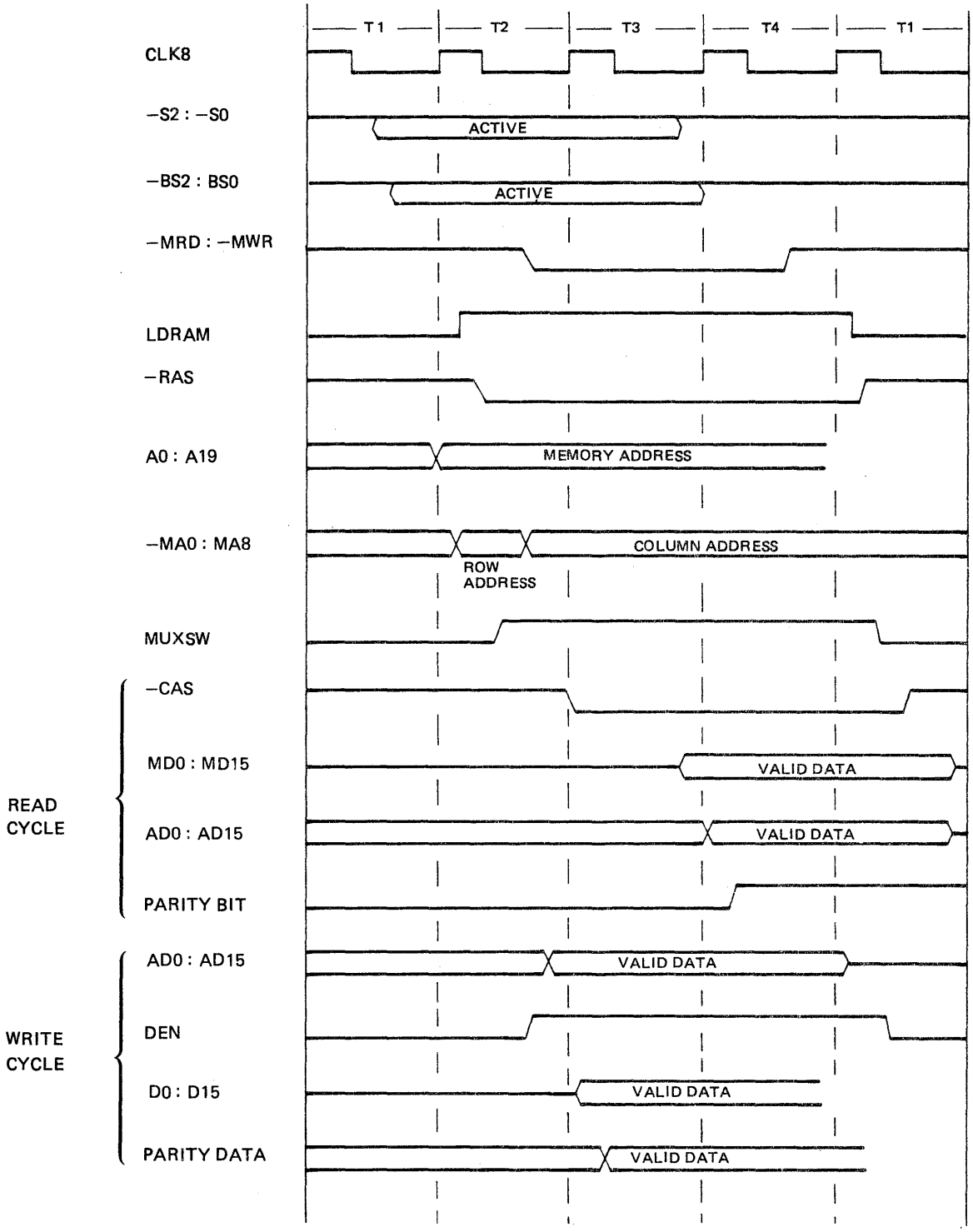


Fig. 2-19 Memory Timing Diagram

MEMORY REFRESH

Refresh cycles are accomplished by programming channel 1 of the timer to request periodically a "dummy" DMA transfer. This action creates a memory read cycle which refreshes dynamic RAM both on the motherboard and expansion boards.

Timer channel 1 output periodically gives rise to DREQ0 (Data Request Signal). The DMA Controller (DMAC) then outputs DMAHRQ (hold request) to the 8086 CPU. When the XHA (hold acknowledge) is asserted, the DMAC gets control of the system busses and issues a -DACK0 signal to both timer channel 1 and the memory control logic circuitry. At this stage the DMAC also issues the memory read signal -MRD. The signal -MRD active resets the F/F LS112 in the memory control logic putting the signal LDRAM high. This signal together with -DACK0B enable the two row address signals -RASL and -RASH.

The first row to be refreshed depends on the state of the memory address lines MA0-MA8. This address will be sequentially increased as the word count register in the DMAC is decremented. These sequential row addresses are latched with -RASL and -RASH and the internal row locations are hence refreshed.

PARITY GENERATOR AND CHECKER

During a memory write cycle two parity bits are generated by the parity generator and checker and written to memory. These two bits are then read back from memory and checked against data for any parity errors during a memory read cycle. If a parity error exists, a non-maskable interrupt is generated the function of which is to stop the CPU operation immediately.

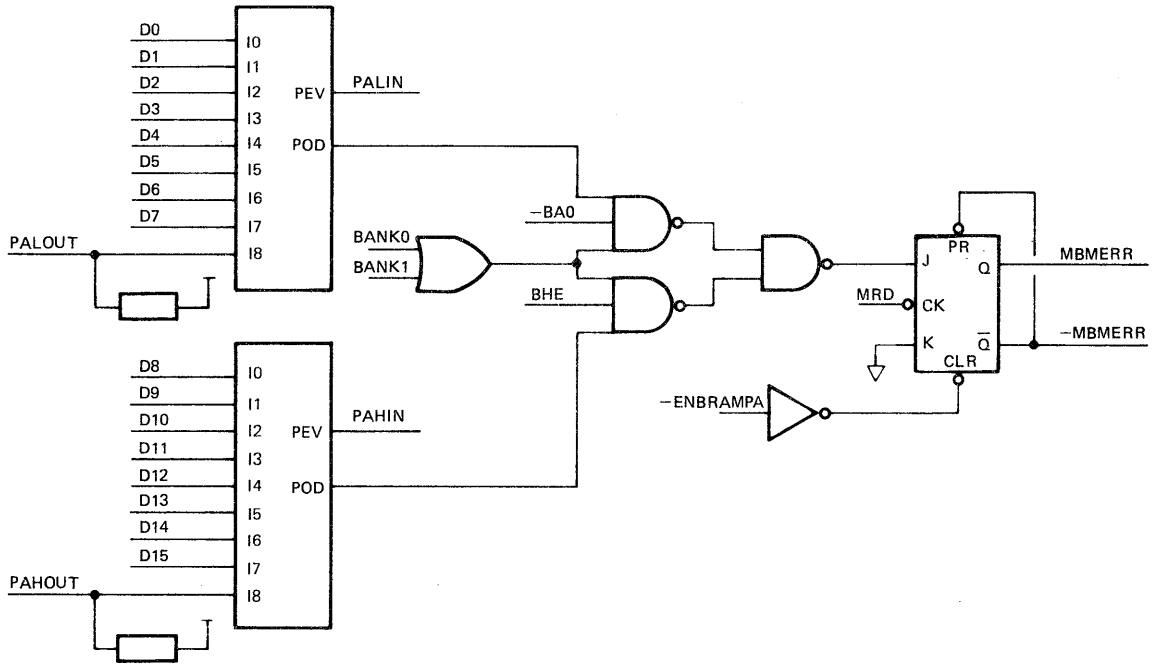


Fig. 2-20 Parity Generator and Checker

During a write memory cycle the fifteen data bits D0-D15 together with the signals PALOUT and PAHOUT (which are high during a write memory cycle) are checked by the parity generators and two bits, PALIN and PAHIN, are generated low or high depending on whether the number of bits is even or odd. These bits are then written into memory.

During a read memory cycle these bits are read back from memory and checked against the data. If there has been no change in the data, then the input to the parity error flipflop is low and so when the MRD signal clocks the F/F, MBERR is low and no parity error is detected. But if there has been a change in the data, the input to the F/F goes high and the F/F is set when clocked. Thus -MBMERR goes low and an 8086 Non-Maskable Interrupt (NMI) is initiated.

PROGRAMMABLE READ ONLY MEMORY

Memory storage for the Power-on Bootstrap and Power-on diagnostics is provided by the EPROMS. Two 8K by 8 bit EPROMS provide 16KB of EPROM. The eight data outputs are connected to the bidirectional data bus MD and the EPROMs address lines are connected to the address bus BA. The chips are controlled by chip enable signal -ROMCE and the output enable signal -MRD. The EPROMs are only accessed by the CPU.

The EPROMs physically occupy addresses F0000 to FFFFF. Following power up or reset the 8086 CPU always begins execution at location FFFF0 where a jump instruction is located. This jump instruction directs the system to the initial program loading routine. When any EPROM location is addressed, PAL14H4 output, PROM, goes active. This in turn enables the PROM chips and also goes on to the wait logic circuitry and produces one wait state. PROM memory cycles require one wait state. The address lines A0-A12 are latched on to the PROMs with the chip enable signal. The desired location in PROM is addressed and the information stored in that location is gated on to the MD lines by the output enable signal -MRD.

INPUT/OUTPUT CHIP SELECT LOGIC

The I/O chip select logic uses a PAL as an address decoder to generate chip select signals from address bits A3 to A9. Chip selection is in accordance with the follow table.

A9	A8	A7	A6	A5	A4	A3	Function	Signal
0	0	0	0	0	0	X	DMA Controller	-8237CS
0	0	0	0	1	0	X	Interrupt Controller	-8259CS
0	0	0	1	0	0	X	Timer	-8253CS
0	0	0	1	0	1	X	SCC	-8530CS
1	1	1	1	1	1	1	ACE	-8530CS
0	0	0	1	1	0	X	I/O Port Select Logic	-IO6X
0	0	0	1	1	1	X	Clock Calendar	-CSXX
0	0	1	0	0	0	X	DMA Segment Register	-DMASEG
1	1	1	1	1	1	0	Floppy Disk Controller	-765CS
0	0	1	0	1	0	X	NMI Register	-NMIREG
1	1	0	1	1	1	1	Parallel Printer Interface	-PNTRCS
0	0	1	1	1	1	X	System I.D. Prom	-SYSID

Note that the DMA controller chip select is not done by th |is PAL but by the logic consisting of the two NOR gates 74LS27 and the AND gate 74LS10.

The chip select signals are also used to generate the enable signal for the PD data bus buffer.

The address decoder is inhibited when the DMA Controller is in control of the system bus.

INPUT/OUTPUT PORT SELECT LOGIC

The I/O port select logic uses a 3 to 8 line decoder LS138 which, when selected by the I/O chip select logic, generates I/O port select signals by decoding address bits A0 to A2. Port selection is in accordance with the following table.

A2	A1	A0	Function	Signal
0	0	0	Keyboard 8041 Data Transfer Read/Write	-I060
0	0	1	Control Port A Read/Write	-I061
0	1	0	Control Port B Read	-I062
0	1	1	Note Used	-I063
1	0	0	Keyboard 8041 Command/Status	-I064
1	0	1	Communications Port Read	-I065
1	1	0	System Configuration Read	-I066
1	1	1	System Configuration Read	-I067

For I/O read | operations the PD bus reads from the selec |ted port. For I/O write op |erations the PD bus writes to the selected |port. Read and wri |te operations are controlled by the Bus Con |troller. The signal -IOR1 controls the direction of the PD bus buffer. When this signal is low (read operation) the direction of data is from the PD bus to the D bus and when it is high (write operation) the direction of data is from the D bus to the PD bus.

SYSTEM CONFIGURATION PORT 66

This port determines the system configuration and is generated by means of the setting of the eight DIP switches in DIPSW-0 on the motherboard. These switches are set according to the following tables:

SW4	SW3	SW2	SW1	BANK 0	BANK 1	EXPANSION BOARD	TOTAL MEMORY
ON	ON	ON	OFF	128K			128K
ON	ON	OFF	ON	128K	128K		256K
ON	ON	OFF	OFF	128K	128K	128K	384K
ON	OFF	ON	ON	128K	128K	256K	512K
ON	OFF	ON	OFF	128K	128K	384K	640K
OFF	ON	ON	ON	512K			512K
OFF	ON	ON	OFF	512K	128K		640K

NOTE : SW4 = ON when 64Kx1 DRAM are used on the motherboard.
SW4 = OFF when 256Kx1 DRAM are installed on the motherboard.

SW NUMBER	OFF	ON
SW8	2764 ROM installed	2732 ROM installed
SW7	Not Used	Not Used
SW6	8530 SCC installed	8250 SCC installed
SW5	8087 NDP installed	No 8087 NDP

SYSTEM CONFIGURATION PORT 67

This port also determines the system configuration and is done through the DIP switches in DIPSW-1 according to the following tables.

SW NUMBER	OFF	ON
SW1	96 TPI floppy drive	48 TPI floppy drive
SW2	Fast start drive	Slow start drive
SW3,SW4	Reserved for HDU	Reserved for HDU
SW5,SW6	Reserved for monitor	Reserved for monitor

SW6	SW5	Display Type
OFF	OFF	IBM Monochrome Display
OFF	ON	80 x 25 Line Character Matrix
ON	OFF	40 x 25 Line Character Matrix

SW8	SW7	No. of Drives
ON	ON	1
ON	OFF	2
OFF	ON	3
OFF	OFF	4

I/O ADDRESS MAP

Address	Usage
000-00F	DMA Controller, 8237A-5
020-021	Interrupt Controller, 8259A
040-043	Timer, 8253-5
050-053	Serial Communication Controller, Z8530
060-063	Emulate 8255A-5 PPI
064	Keyboard, 8041A Command/Status
066-067	System Configuration
070-07F	Clock/Calendar Chip
0F0-0FF	System ID PROM
210-217	System Expansion Box
320-32F	Hard Disk Drive Controller
378-37F	Parallel Printer
380-38F	SDLC Communications
3C0-3CF	Reserved
3D0-3DF	Display Controller
3F0-3F7	Diskette Drive Controller
3F8-3FF	Asynchronous Communications Controller, 8250

SYSTEM MEMORY ADDRESS MAP

<table>
<tr><th>Address</th><th colspan="2">Function</th></tr>
<tr><td>00000
↓
1FFFF</td><td>256 KB Read/Write Memory on Motherboard

128 KB on BANK0, 64Kx1 DRAMs</td><td rowspan="4">640 KB Read/Write Memory on Motherboard

512 KB on BANK0, 256Kx1 DRAMs</td></tr>
<tr><td>20000
↓
3FFFF</td><td>128 KB on BANK1, 64Kx1 DRAMs</td></tr>
<tr><td>40000
↓
5FFFF</td><td>384 KB Read/Write Memory on Memory Expansion Board

128 KB on BANK0, 64Kx1 DRAMs</td></tr>
<tr><td>60000
↓
7FFFF</td><td>128 KB on BANK1, 64Kx1 DRAMs</td></tr>
<tr><td>80000
↓
9FFFF</td><td>128 KB on BANK2, 64Kx1 DRAMs</td><td>128 KB on BANK1, 64Kx1 DRAMs</td></tr>
<tr><td>A0000
↓
AFFFF</td><td colspan="2">64 KB Reserved</td></tr>
<tr><td>B0000
↓
B7FFF</td><td colspan="2">32 KB Reserved</td></tr>
<tr><td>B8000
↓
BFFFF</td><td colspan="2">32 KB Display Controller Video Memory</td></tr>
<tr><td>C0000
↓
FBFFF</td><td colspan="2">240 KB Read Only Memory - Expansion and Control</td></tr>
<tr><td>FC000
↓
FFFFF</td><td colspan="2">16 KB Boot Read Only Memory</td></tr>
</table>

TIMER

The system uses the INTEL 8253 programmable interval timer with three independent counters. The three counters are used as follows:

- Counter 0 to provide a fixed frequency timing signal for the Operating system to keep track of timing dependent activities. In fact it provides a real-time clock.

- Counter 1 is used to time and request refresh cycles from the DMA counter.

- Counter 2 is used to support the tone generation for the audio speaker.

The function of the timer is that of a general purpose, multi-timing element that can be treated as an array of I/O ports in the system software.

The timer block diagram is shown in figure 2-22. It is made up of the following blocks:

- Data Bus Buffer
- Read/Write Logic
- Control Word Register
- Counters 0,1,2

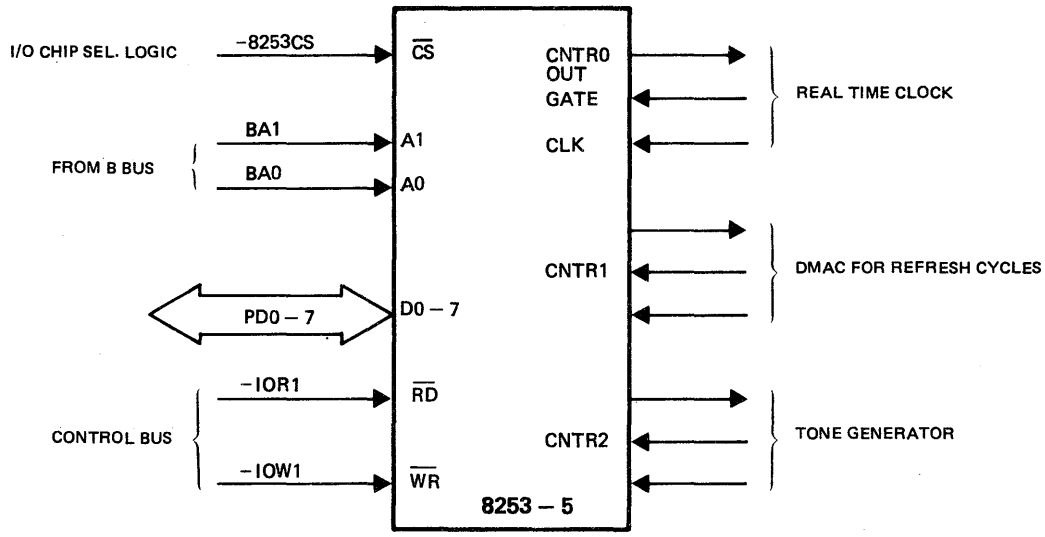


Fig. 2-21 Timer System Interface

Data Bus Buffer

This tri-state bidirectional, 8 bit buffer is used to interface the 8253 to the PD Bus. The Data Bus Buffer has three basic functions;

- Programming the MODES of the 8253
- Loading the count registers
- Reading the count values

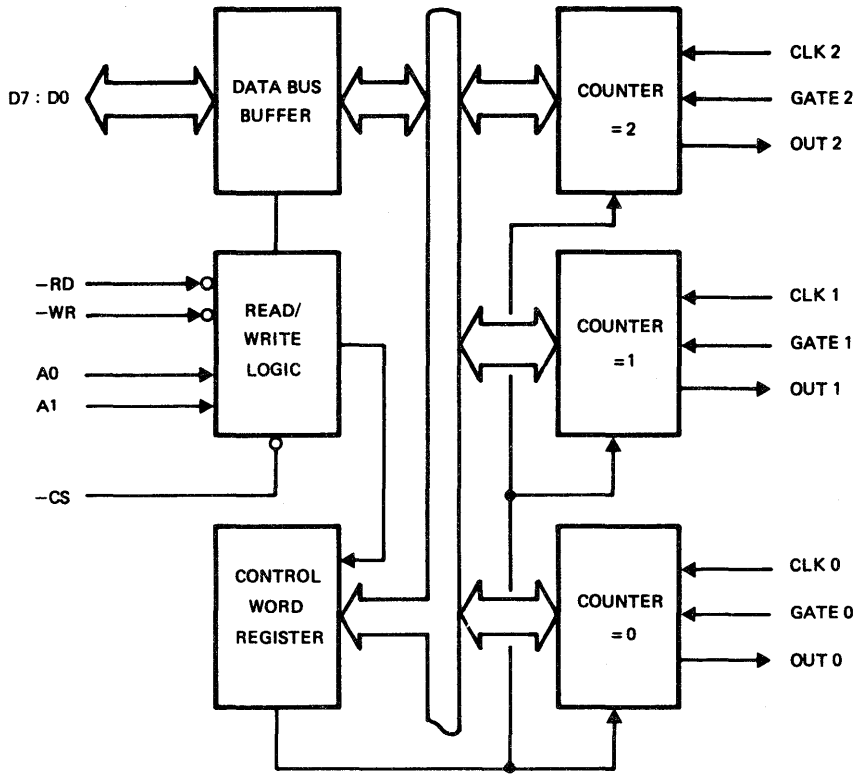


Fig. 2-22 Timer Block Diagram

Read/Write Logic

The Read/Write logic accepts inputs from the system bus and in turn generates control signals for overall device operation. It is enabled or disabled by -8253CS so that no operation can occur to change the function unless the device has been selected by the system logic.

Lines BA0 and BA1 select one of the three counters to be operated on and address the control word register for mode selection.

Control Word Register

The Control Word Register is selected when both BA0 and BA1 are high. It then accepts information from the data bus buffer and stores it in a register. The information stored in this register controls the operational MODE of each counter, selection of binary or BCD counting and the loading of each count register. The Control Word Register can only be written into. No read operation of its contents is available.

Counters 0,1,2

These three functional blocks are identical in operation. Each counter consists of a single 16 bit, pre-settable DOWN counter. The counter can operate in either binary or BCD and its input gate and output are configured by the selection of MODES stored in the Control Word Register.

The counters are fully independent and each can have separate Mode configuration and counting operation, binary or BCD.

REAL TIME CLOCK AND CALENDAR

The clock/calendar chip is used to provide real time such as seconds, minutes, hours, day of week, days, months etc. for the system. A 3 volt low power standby battery is connected to it so as to ensure that the real time is kept even when the personal computer is turned off. The chip used is the MM58174AN.

The block diagram shown in figure 2-23 shows the structure of the clock/calendar chip.

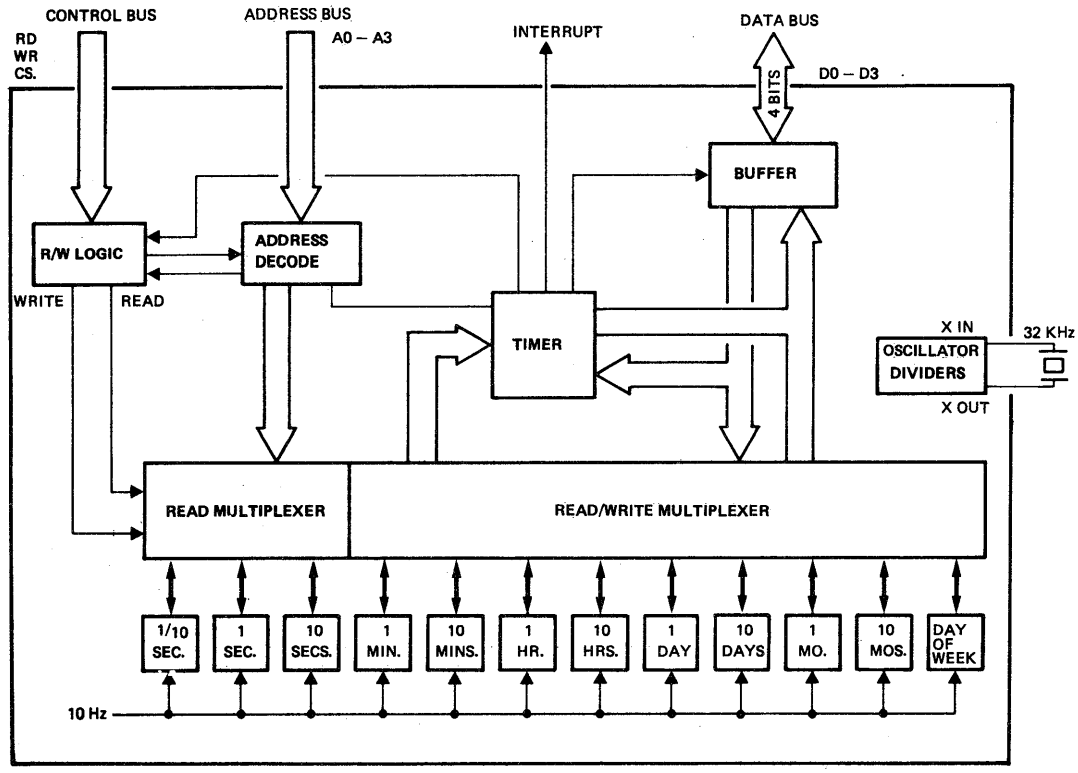


Fig. 2-23 Clock/Calendar Chip Block Diagram

Connected externally to pins 15 and 14 of the clock calendar chip is the crystal oscillator (32.768KHz). A 15pF capacitor is also connected to pin 14 and a single 6-36pF variable capacitor to pin 15. This capacitor is used to fine tune the oscillator.

The interrupt output of this chip is used in the system.

When writing the initial parameters to this chip, during the initialization period, the write cycle is longer than the normal -IOW1 cycle. Hence signal -IOW1 is ORed with a signal -QD which is the output of a counter. In this way the write cycle to this chip is extended and all the required parameters may be written into it. The same thing applies to the read cycle.

SPEAKER INTERFACE

The sound system has a small, permanent magnet speaker that may be driven from the following sources:

- A program controlled output bit SPKRDATA from control port A which may be toggled to generate a pulse train.

- Timer counter 2 output. This counter is programmed in mode 3 as a square wave generator or as a programmed controlled bit. The signal TMR2GSPKR from control port A goes to the gate input of counter 2 and is modulated with the 1.2MHz clock to generate the counter 2 output signal 530UT.

The speaker drive system is shown below:

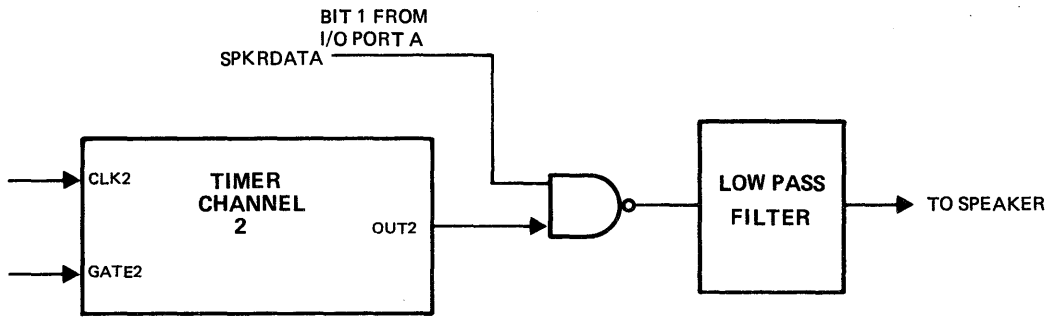


Fig. 2-24 Speaker Interface

KEYBOARD INTERFACE

The Keyboard Interface uses an Intel 8041 microcomputer as a keyboard controller to convert the system parallel data into serial data for transmission to the keyboard and vice versa. The 8041 controller is a single chip microcomputer with 1K bytes of internal ROM, 64 bytes internal RAM, 16 I/O ports and an 8 bit data bus.

The keyboard controller monitors the I/O ports 12 and 13. A low signal on port 12 generates a keyboard hardware reset which sets the KBCLOCK low for at least 50ms. The 8041 monitors this input every 2ms. Port 13 signal, KB/-TYP is not used when a PC keyboard is connected.

The address input BA2 indicates to the controller whether the byte transfer is data (BA2 is low) or a command (BA2 is high).

COMMUNICATION BETWEEN 8041 AND KEYBOARD

The communication between the 8041 keyboard controller and the keyboard itself is in a bit asynchronous format utilizing two signals namely KBCLOCK and KBDATA.

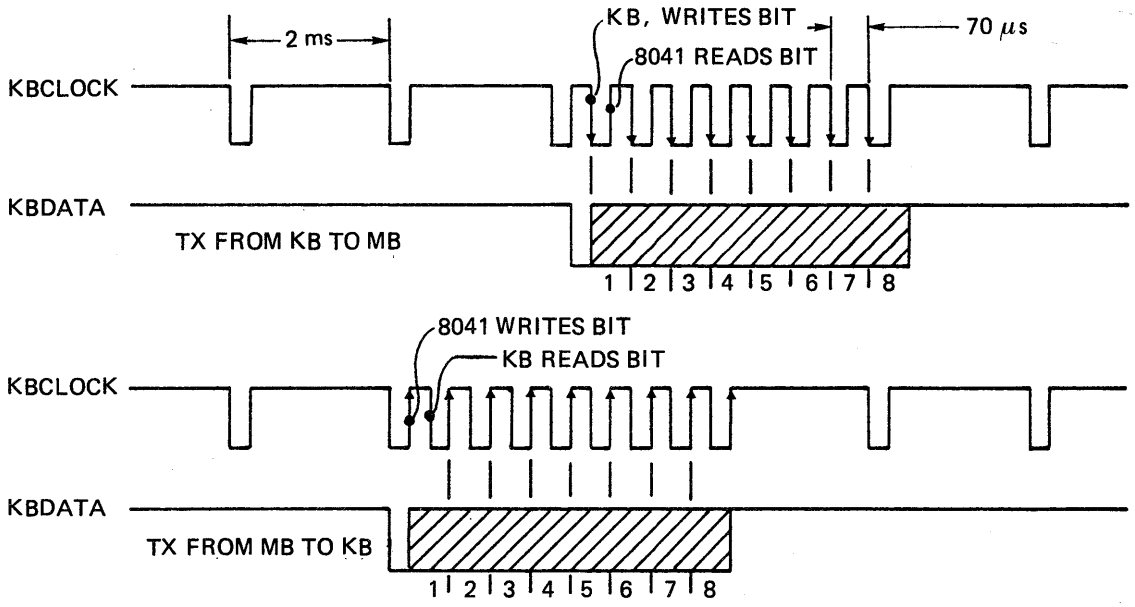


Fig. 2-25 Communication between 8041 and Keyboard

The KBCLOCK signal is generated by the 8041 chip and it establishes the transmission rate. The KBDATA signal is bidirectional and the direction of transmission is set according to the condition of KBDATA at the first falling edge of KBCLOCK.

Every 2ms the 8041 sets the KBCLOCK signal low and if a code is to be sent to the keyboard it also sets the KBDATA signal low. This condition produces a reception keyboard interrupt routine. The 8041 then sends 8 clocks on the KBCLOCK signal to the keyboard. The 8041 writes a bit on the rising edge of KBCLOCK while the keyboard reads the bit on the falling edge of the clock signal.

If no code is to be transmitted to the keyboard, KBDATA remains high and a transmission keyboard interrupt routine is generated. Now, if the keyboard wants to transmit a code to the interface, KBDATA goes low . The 8041 reads the signal KBDATA and, if this is low, starts a reception interrupt routine. This routine sends 8 clocks to the keyboard and on every falling edge of KBCLOCK the keyboard writes one bit while the 8041 controller reads the bit on every rising edge.

The keyboard interface signals are made available through a 9-pin, D-shell connector. This connector protrudes through the rear of the basic unit to facilitate the connection of the keyboard cable.

The keyboard interface signals are as follows:

KBDATA	Keyboard Serial Data
KBCLOCK	Keyboard Clock
-RESET1	Keyboard Reset
P12	+12V Power Supply
GND	Ground

TMS7000 BOARD

This board which is based on the TMS7000 microprocessor emulates the 8041 keyboard interface. It is present on some of the system motherboards to replace the 8041 microprocessor and plugs into the socket which is usually occupied by the 8041.

The TMS7000 is a single chip microprocessor with 128 bytes of internal RAM, 16 I/O lines and an 8 bit data bus.

The circuitry on this board also consists of a 1 KB EPROM, three 8-bit latches and associated control logic required for the control of the data bus. The keyboard controller communicates with the system through the 8 bit data bus. It then converts the parallel data into serial data or vice-versa to communicate with the keyboard itself through the clock in/out pins and the data in/out pins.

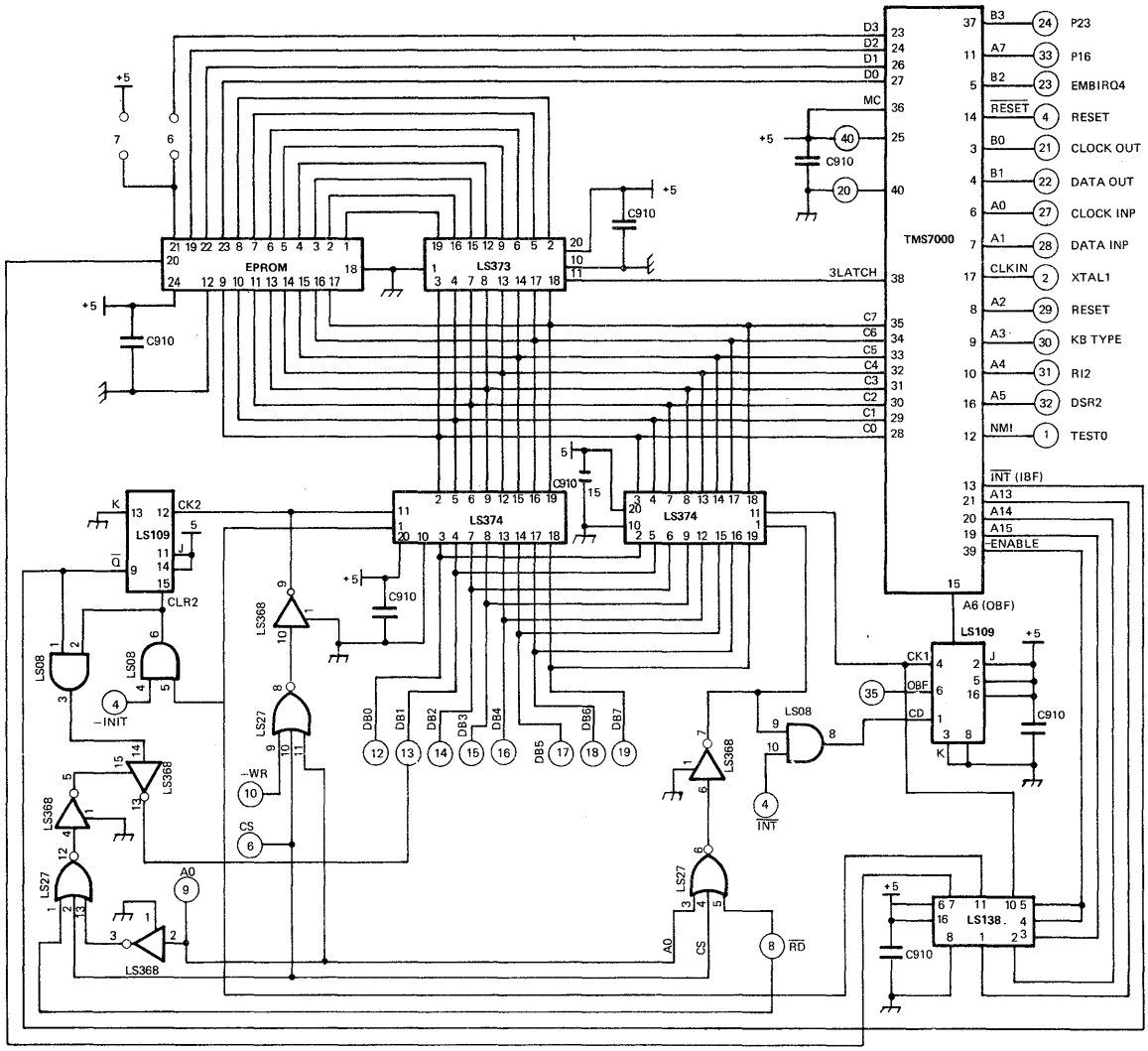


Fig. 2-26 TMS7000 Board Circuitry

MINI-FLOPPY DISK INTERFACE

The Mini-Floppy Disk Interface is resident on the motherboard and provides two major functions:

- It provides logic and control circuitry necessary to control and record data onto, or read data from the 5.25 inch mini-floppy disk.

- It initially formats new disks (identifies sectors and tracks).

This circuitry keeps track of head positioning and, acting in response to commands issued by the CPU under Operating System (OS) control, selects the requested sector and track for each read or write operation. This circuitry is used for double-density, MFM-coded disk drives as well as for single density FM-coded disk drives. It uses write precompensation with an analogue phase-lock loop for clock and data recovery. The controller used is an uPD765 controller. The disk drive parameters are programmable. In addition, this circuitry supports the disk drive's write protect feature. This circuitry uses Direct Memory Access (DMA) for record data transfers. An interrupt is also sent to the CPU to indicate that an operation has been completed and that a status condition requires CPU attention.

PRINCIPLES OF OPERATION

Figure 2-27 shows the Mini-Floppy Disk Interface Block diagram. The major elements of this interface are:

- Mini-Floppy Disk Controller Chip (uPD765)

- Clock and Timing Circuit

- Digital Control Port and Decoder

- Write Precompensation Circuit

- Data separator

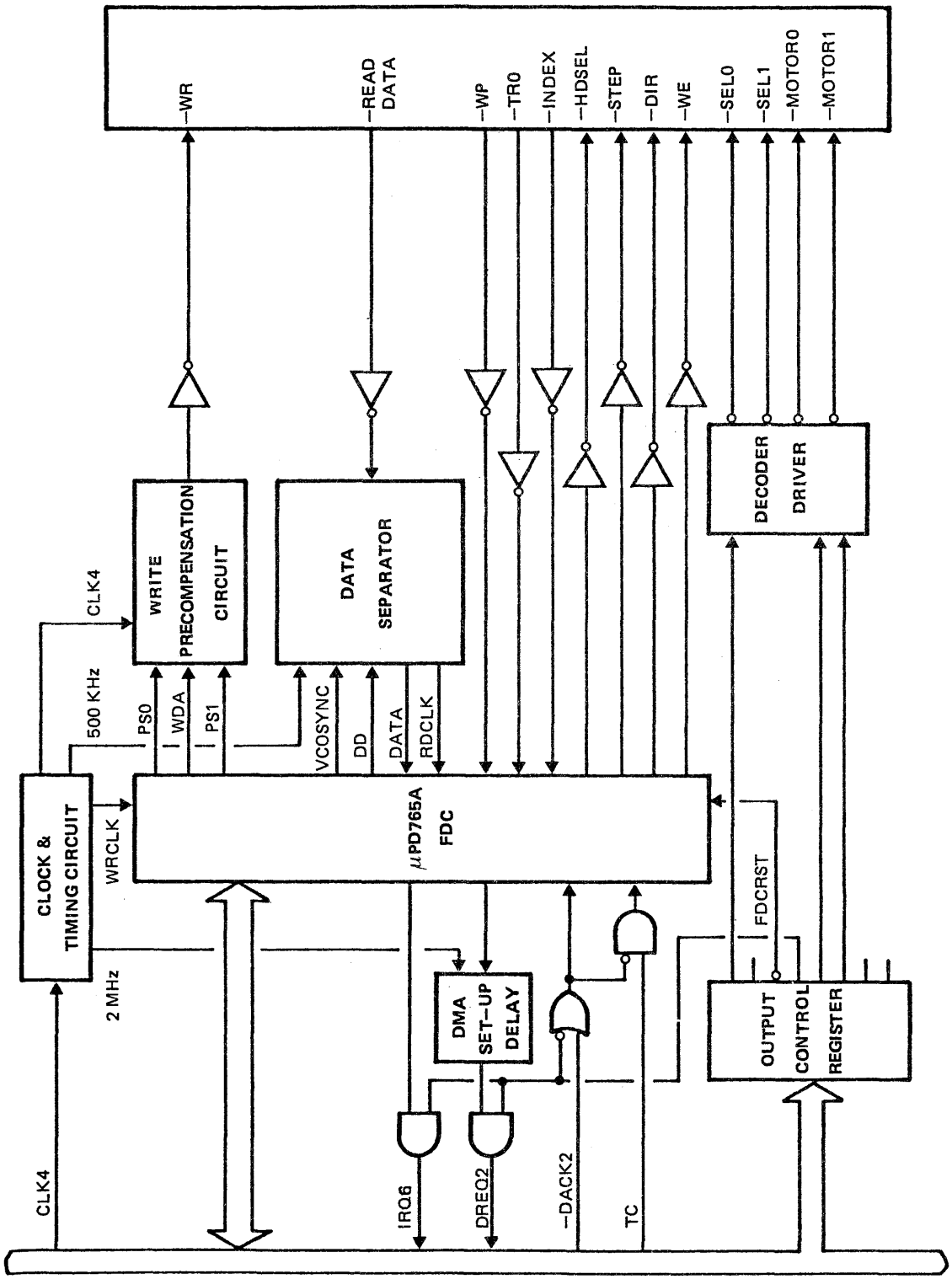


Fig. 2-27 Mini-Floppy Disk Interface Block Diagram

FLOPPY DISK CONTROLLER

The controller used is the uPD765. This LSI chip contains the circuitry and control functions for interfacing the system CPU to 4 Mini-Floppy Disk Drives. It is capable of supporting either IBM 3740 single density format (FM) or IBM 34 Double Density format (MFM) including double sided recording. The controller operates in a DMA mode during data transfers. In this way the CPU need only load the command required into the controller and all data transfers occur under the control of the uPD765 and the DMA Controller. The uPD765 supports 15 separate commands. A block diagram of the uPD765 is shown in figure 2-28.

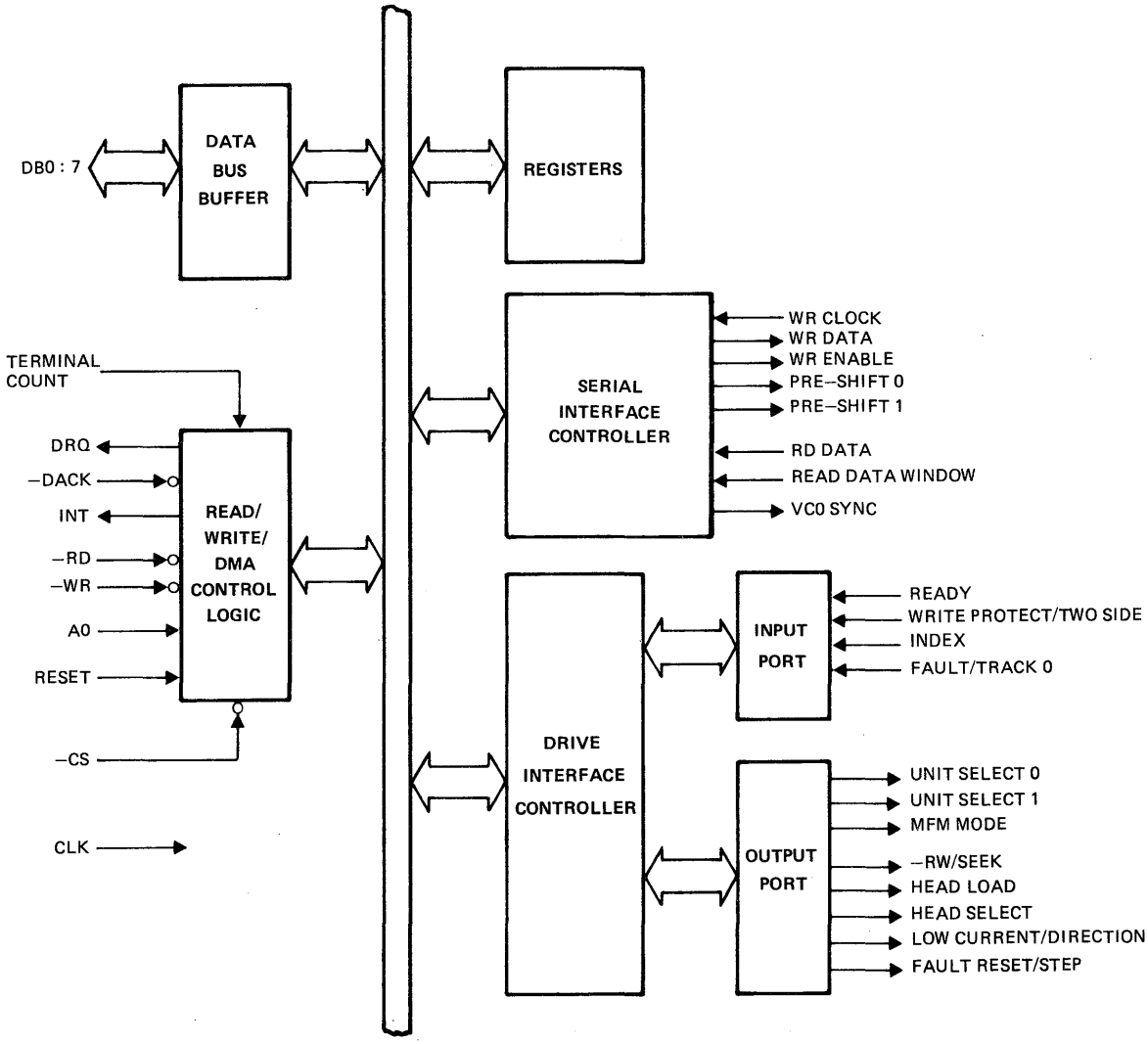


Fig. 2-28 uPD765 Block Diagram

uPD765 Controller Pin Functions

The following is a description of the pin functions for this controller.

RST Reset: This signal places the FDC in an idle state and resets output lines to the floppy disc drives.

-RD Read: This is the control signal for transfer of data from FDC to the data bus and is active low.

-WR Write: This is the control signal for transfer of data to the FDC via the data bus and is active low.

-CS Chip Select: This pin selects the FDC when it is low allowing -RD and -WR pins to be enabled.

A0 Data/Status Register Select: This signal selects the data register (A0 high) or the status register(A0 low) contents of the FDC to be sent to the data bus.

DB0-DB7 Data Bus: These lines constitute the bi-directional 8 bit data bus.

DRQ Data DMA Request: This signal is high when a DMA request is made by the FDC.

-DACK DMA Acknowledge: This input signal indicates that a DMA cycle is active when low and the controller is performing a DMA transfer.

TC Terminal Count: When high, this signal indicates the termination of a DMA transfer. It terminates data transfer during read/write/scan command in a DMA or interrupt mode.

IDX Index: This pin indicates the beginning of a disk track.

CLK Clock: This input clock is a single phase 8 MHz squarewave clock.

WCK Write Clock: This clock indicates the the write data rate to the floppy disk drive which is 500KHz in FM mode and 1MHz in MFM mode. The pulse width is 250ns for both modes.

RDW Read Data Window: This is generated by the phase-lock loop and is used to sample data from the floppy disk drive.

RDD Read Data: This is the read data from the floppy disk drive containing clock and data bits.

VCO VCO Sync: This signal inhibits VCO in phase-lock loop when high and enables VCO when low.

WE Write Enable: This signal enables write data to the floppy disk drive.

MFM MFM Mode: This signal selects between MFM mode when high and FM mode when low.

HD Head Select: This selects head 1 when high and head 0 when low.

US0, US1 Unit Select: These pins are not connected.

WDA Write Data: This line carries the serial clock and data bits to the disk drive.

PS1, PS0 Precompensation: These two lines indicate the write precompensation status during MFM mode. They determine early, normal or late times.

FLT/TR0 Fault/Track 0: This line sense the drive fault condition in read/write mode and track 0 in seek mode.

WP/TS Write Protect/Two-Side: This signal senses the write protect status in read/write mode and the two media in seek mode.

RDY Ready: This signal indicates that the drive is ready to send or receive data.

HDL Head Load: This signal causes the read/write head in the disk drive to load onto the diskette.

FR/STP Fit Reset/Step: This signal resets fault flip flop in disk drive when in read/write mode and contains step pulses to move head to another cylinder in seek mode.

LCT/DIR Low current/Direction: This signal lowers the write current on inner tracks during a read/write mode and determines the direction the head will step during a seek mode. A fault reset pulse is issued at the beginning of each read or write command prior to the occurrence of the head load signal.

-RW/SEEK Read Write/Seek: This line selects the seek mode when high and the read/write mode when low.

Figure 2-29 shows the uPD765 - M24/M21 system configuration.

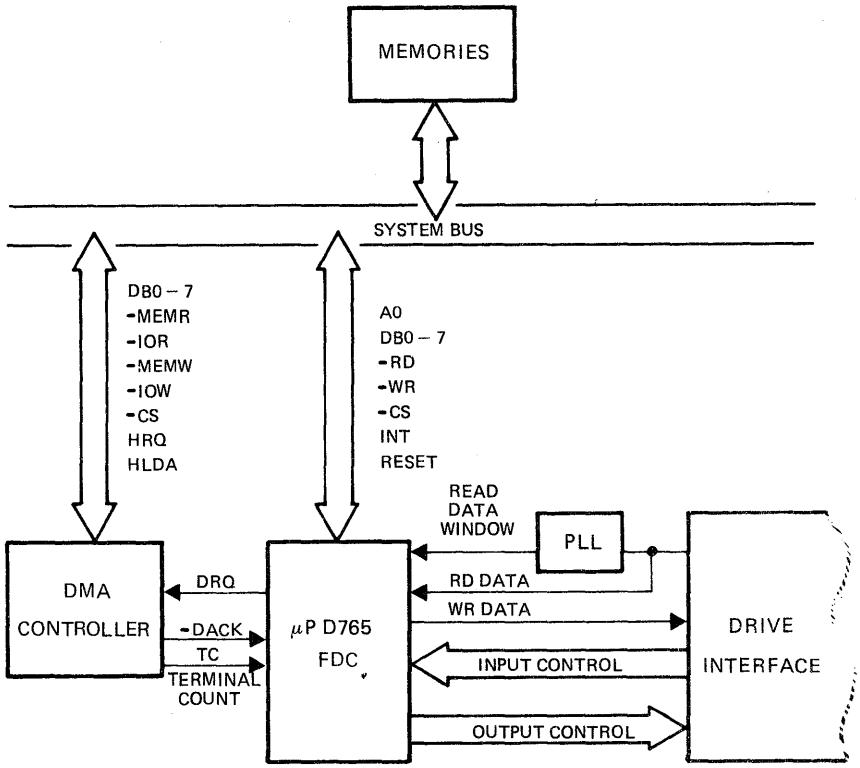


Fig. 2-29 uPD765 - M24/M21 System Configuration

WRITE PRECOMPENSATION CIRCUITRY

The object of this circuitry is to cause the Write Data output from the uPD765A to be written early, late or on time on the mini-floppy disk drive. The write precompensated data stream input to the disk drive is WD.

Write Precompensation is a technique mostly used in MFM recording. The following is a brief explanation of what precompensation is all about. Due to the decreasing radius on the physical surface of the disk, the inside tracks have less circumference and therefore exhibit an increase in recording flux density over the outside tracks. This increase in flux density aggravates a problem in magnetic recording known as 'dynamic bit shift'.

Dynamic bit shift comes about as the result of one bit on the disk (a flux reversal) influencing an adjacent bit. The effect is to shift the leading edge of both bits closer together or further apart than recorded. The net result is that enough jitter is added to the recorded data on the inside tracks to make them harder to recover without error. In any event, there is a method which can be applied to reduce the effect of this shift on the data called 'write precompensation'. Precompensation is a way of predicting which direction a particular bit will be shifted and intentionally writing that bit out of position in the opposite direction

to the expected shift. This is done by examining the status during MFM recording to produce early, late and normal timings. Signals PS1 and PS0 as seen below determine these timings.

PS0	PS1	
0	0	Normal
0	1	Late
1	0	Early
1	1	Invalid

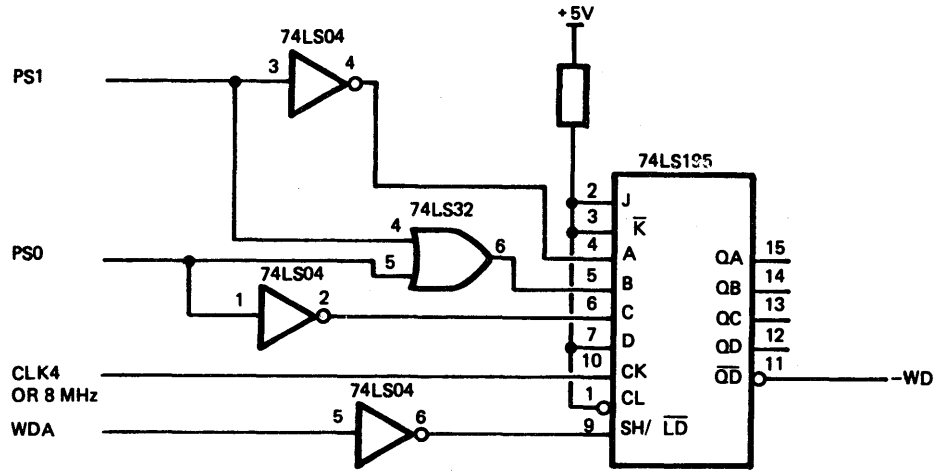


Fig. 2-30 Write Precompensation Circuitry

The write precompensation circuitry is done by the three floppy disk controller outputs WDA, PS0, and PS1 which are gated into the LS195 shift register. This shift register adds a 500ns delay to the write data. If the FDC signals 'early shift' then this delay time is reduced to 250ns by loading the data into cell 0 rather than cell 1. If the FDC signals 'late shift' then data is loaded into cell 2 causing a delay of 750ns. The data is then clocked serially out of this shift register as write data -WD.

DIGITAL CONTROL PORT AND DECODER

This circuitry receives an instruction from the CPU via the PD bus at latch LS273. The outputs of this latch are then decoded by the four 74LS38 gates to generate the signals signals -SEL0,-SEL1 to select the required mini-floppy disk drive and -MOTOR0,-MOTOR1 to drive the mini-floppy motor selected. The PD0-7 inputs are latched to the outputs of the LS273 when one of the signals -IOW and -SELRG makes a low to high transition.

The control signals HD (select heads), STEP, DIR (direction), WE (write enable), WD (write data) which are generated by the floppy disk controller pass through the decoder 74LS38 to become -HDSEL, -STEP, -DIR, -WE and -WD respectively.

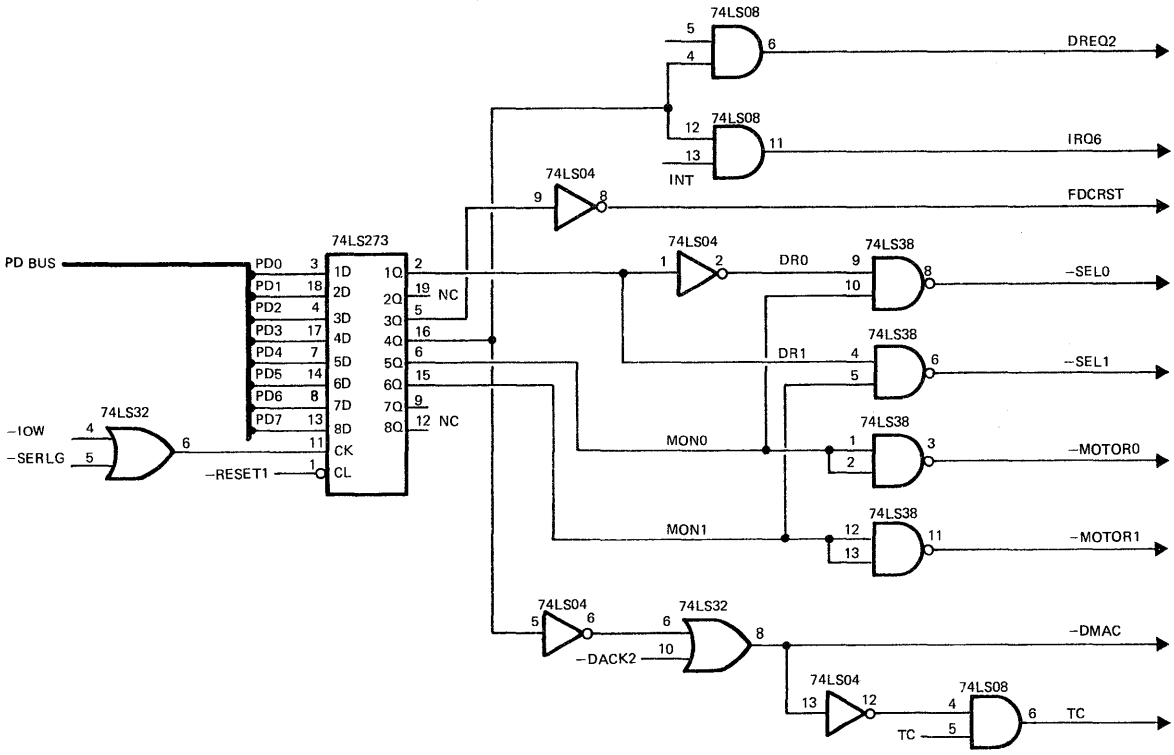


Fig. 2-31 Control Port and Decoder Circuitry

CLOCK AND TIMING CIRCUITRY

This circuitry consists mainly of two divider counters which convert the CLK4 into a 2MHz clock, a 1MHz clock, a 500KHz clock and a 250KHz clock. The 500KHz clock is used in the data separator circuitry. The 2MHz clock is used to latch the DMA request and send it to the DMAC as DREQ2. The four clocks are gated to generate the write clock, WRCLK, used by the FDC.

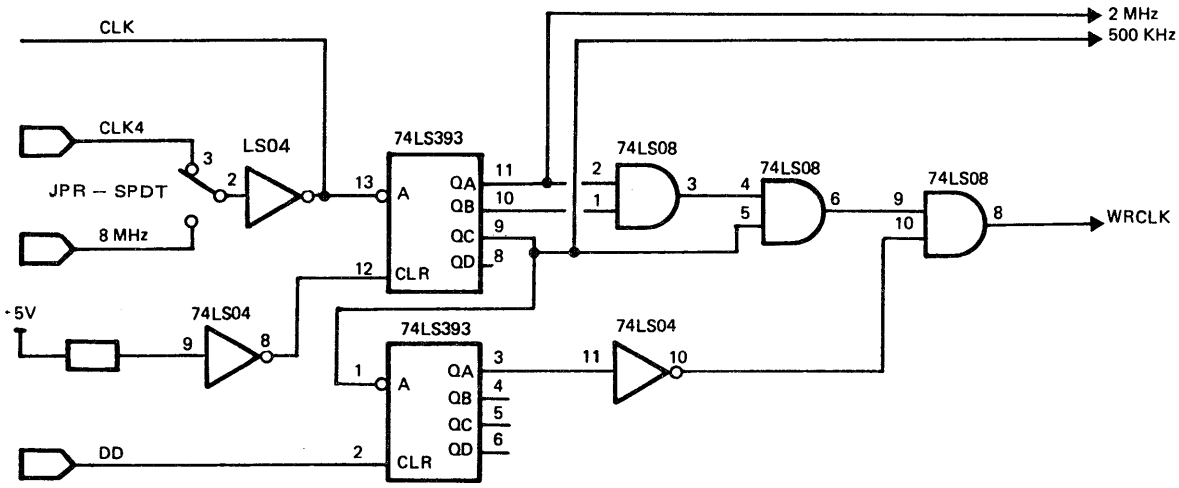


Fig. 2-32 Timing Circuitry

DATA SEPARATOR

Data is recorded on to the disk using two different techniques, FM and MFM. In order to provide maximum data recording density, the Modified Frequency Modulation (MFM) technique is mostly used. This technique requires clock bits to be recorded only when two successive data bits are missing in the serial data stream. This reduces the total number of bits required to record a given amount of information on the disk. The fact that clock bits are not recorded with every data bit cell requires circuitry to synthesize clock bit timing when clocks are missing and to synchronize to clock bits when they are present.

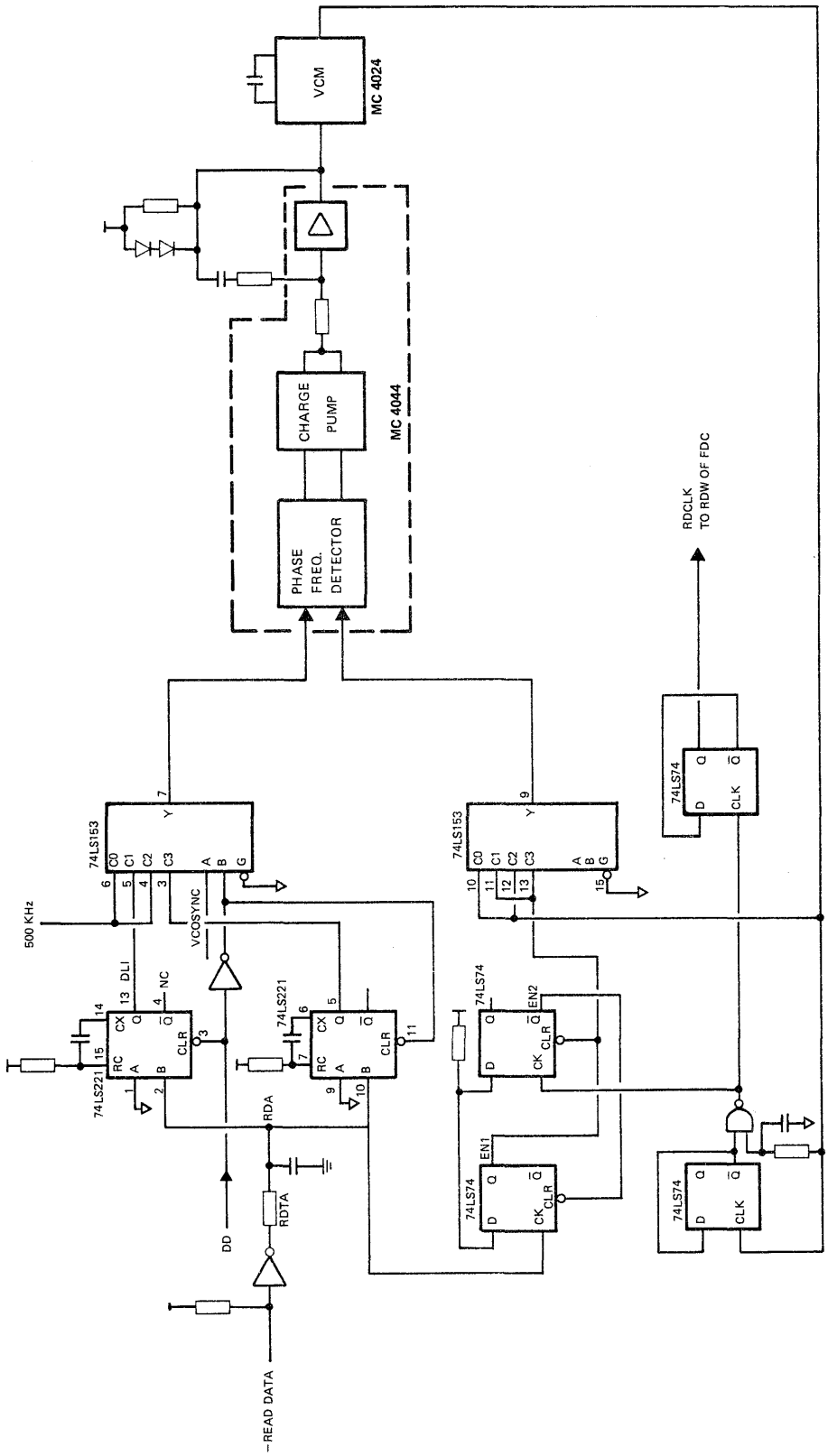


Fig. 2-33 Data Separator Circuit

This circuitry contains an analogue phase-lock loop. Its function is to provide a read 'clock' locked to the incoming data which follows small variations in the motor speed but does not respond to the jitter caused by magnetic recording distortions or noise.

The Phase-Lock Loop circuitry consists of the MC4044 chip which contains a phase frequency detector, a charge pump and an amplifier. It provides an error signal which is proportional to the phase difference between its inputs and controls the output VCO frequency of the MC4024. In fact its output goes to a voltage-controlled multivibrator, MC4024, whose output frequency varies according to the input voltage and thus the signal RDCLK is varied accordingly.

The clock output from the MC4024 is divided by two as required by the uPD765 controller. Another flip flop divides the frequency down by another factor of two in the FM mode. The other two flip flops form part of a timing filter to pass the correct time pulses to the phase detector. Fig. 2-34 is a timing diagram for this section of circuitry.

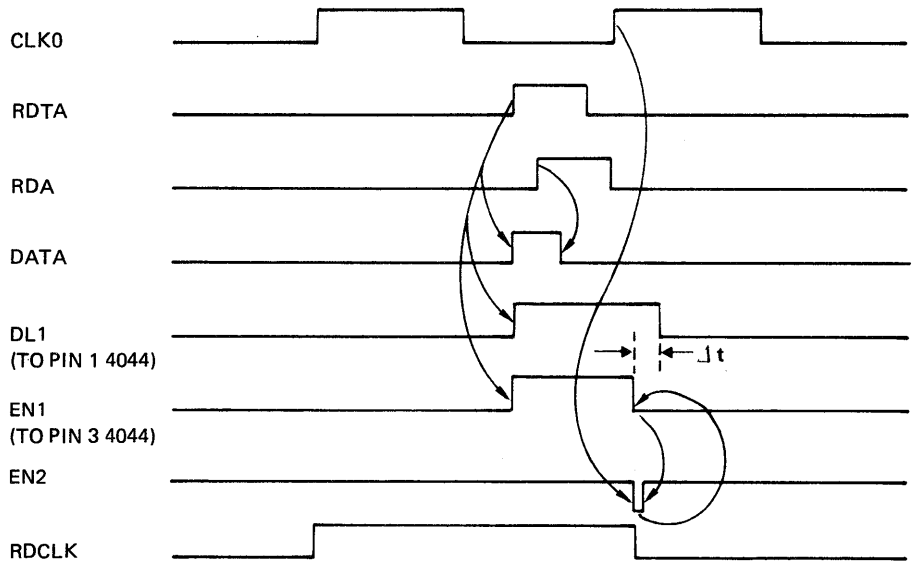


Fig. 2-34 Data Separator Timing in MFM Read Mode

The 74LS221 flip flop provides a 1us delay to the read data. This causes a 90 degrees phase difference between the data and the RDCLK as is required by the 765 controller. In the case of FM single density recording this delay is 2us. The timing filter needs this delay in order to know the presence of the data pulses in advance.

The 74LS153 multiplexer is used to switch between single and double density and between read and wait modes. In the wait mode the phase lock loop is locked to the 500KHz.

PARALLEL PRINTER INTERFACE

A Centronics-like parallel printer interface is provided on the motherboard. A block diagram of this interface is shown in Figure 2-35.

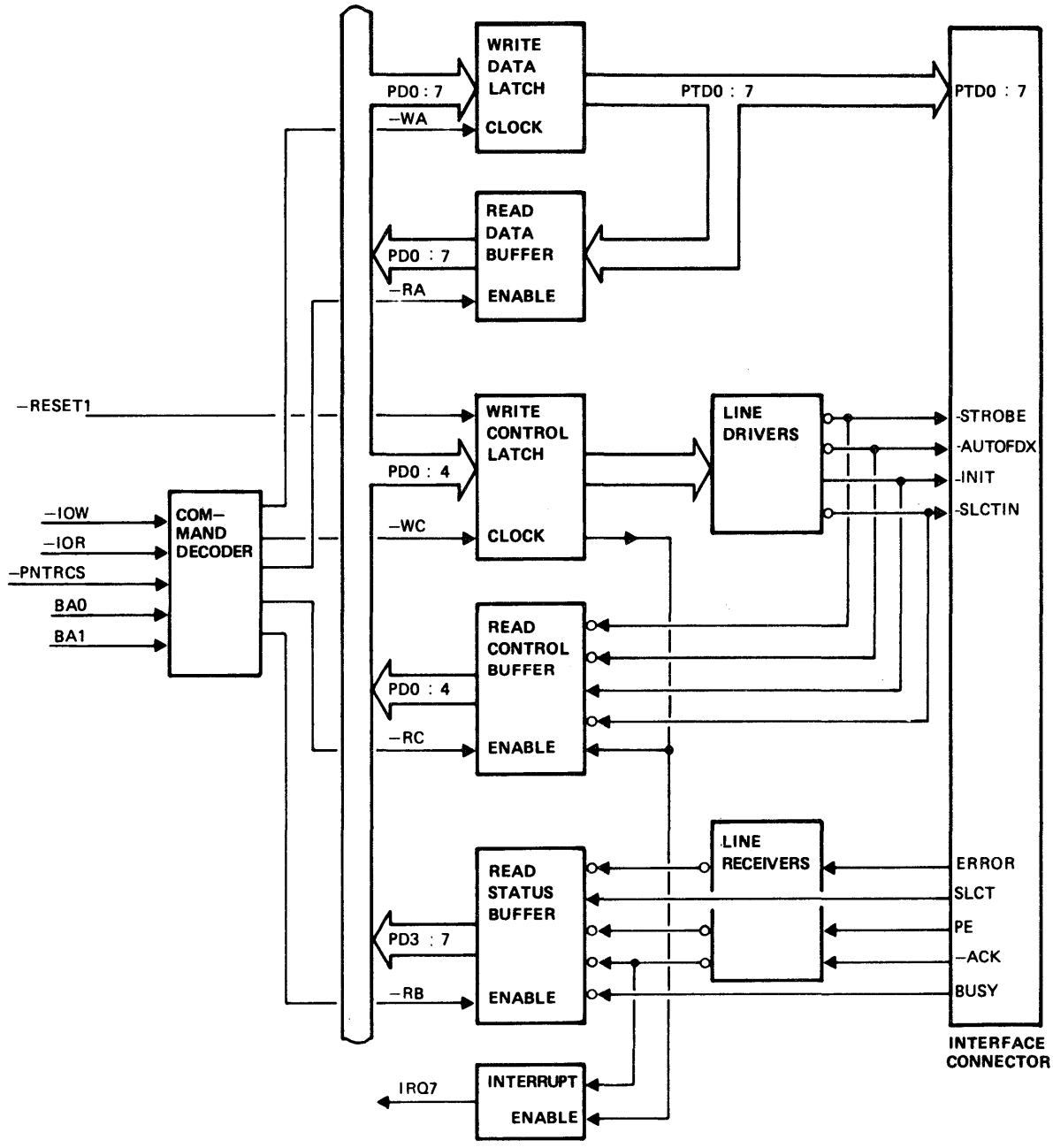


Fig. 2-35 Parallel Printer Interface Block Diagram

The interface uses a dual 2 to 4 line, LS155, decoder to generate read and write signals in response to I/O instructions by decoding address bits BA0, BA1 and the I/O read and write signals, -IOR and -IOW.

Printer output instructions generate write signals -WA and -WC that enable data into the two output latches, LS373 and LS174. Signal -WA writes data bits PD0 to PD7 into one latch in order to generate the printer data bus and signal -WC writes data bits PD0 to PD5 into the other latch in order to generate the interface control signals strobe -STROBE (D0), Auto Feed -AUTOFDX (D1), Initialize Printer -INIT (D2), Select Input -SLCTIN (D3) and a CPU interrupt. The CPU interrupt is generated on the falling edge of the printer Acknowledge signal if the Interrupt Request bit (D4) is written high.

Printer input instructions generate read signals -RA, -RB and -RC that enable data on the printer interface to be written onto the databus. Signals -RA and -RC enable input buffers that read back the contents of the two output latches into the same bit positions and signal -RB enables input buffers that read the printer status bits namely Acknowledge -ACK (D6) , Busy BUSY (D7), Paper End PE (D5), Error and Select SLCT (D4).

The parallel Printer interface signals are made available through a 25-pin, D-shell connector. This connector protrudes through the rear panel of the Basic Unit to facilitate the connection of the printer cable.

The Parallel Printer Interface signals are as follows:

-STROBE	A low level pulse used to transfer character data from the interface to the printer.
PTD0 to PTD7	Data lines that carry the character data from the interface to the printer.
-ACK	A low level indicates that the current character has been accepted by the printer and the printer is ready to receive new data.
BUSY	A high level indicates that the printer cannot accept new data.
PE	Paper End. A high level indicates that the printer is out of paper.
SLCT	This signal indicates that the printer is in the selected condition.
-AUTOFDX	A low level instructs the printer to feed one line of paper automatically after printing.
ERROR	A low level indicates that the printer is in the error condition.
-INIT	A low level pulse that resets the printer to its initial state and clears the printer buffer.
-SLCTIN	A low level enables the printer to accept new data.
GND	Ground

SERIAL COMMUNICATION INTERFACE

The serial communication interface uses an 8250 Asynchronous Communications Element (ACE) or a dual channel Z8530 Serial Communications Controller (SCC) to generate EIA RS-232-C interchange circuit signals for data and modem control and timing. Associated line drivers and receivers convert these signals from TTL levels to EIA RS-232-C interchange circuit voltage levels and vice-versa. The interface is fully programmable.

Channel B of the Z8530 SCC is used in conjunction with the Multi Communications Interface expansion board. This board plugs into one of the system expansion slots and is connected to the motherboard by a ribbon cable.

8250 ASYNCHRONOUS COMMUNICATIONS ELEMENT (ACE)

The asynchronous communications element used is an 8250. This performs serial-to-parallel conversion of data on the Received Data circuit, parallel-to-serial conversion of data on the peripheral data bus for transmission on the Transmitted Data circuit and the modem control functions; clear to send, request to send, data set ready, data terminal ready, ring indicator and data carrier detected.

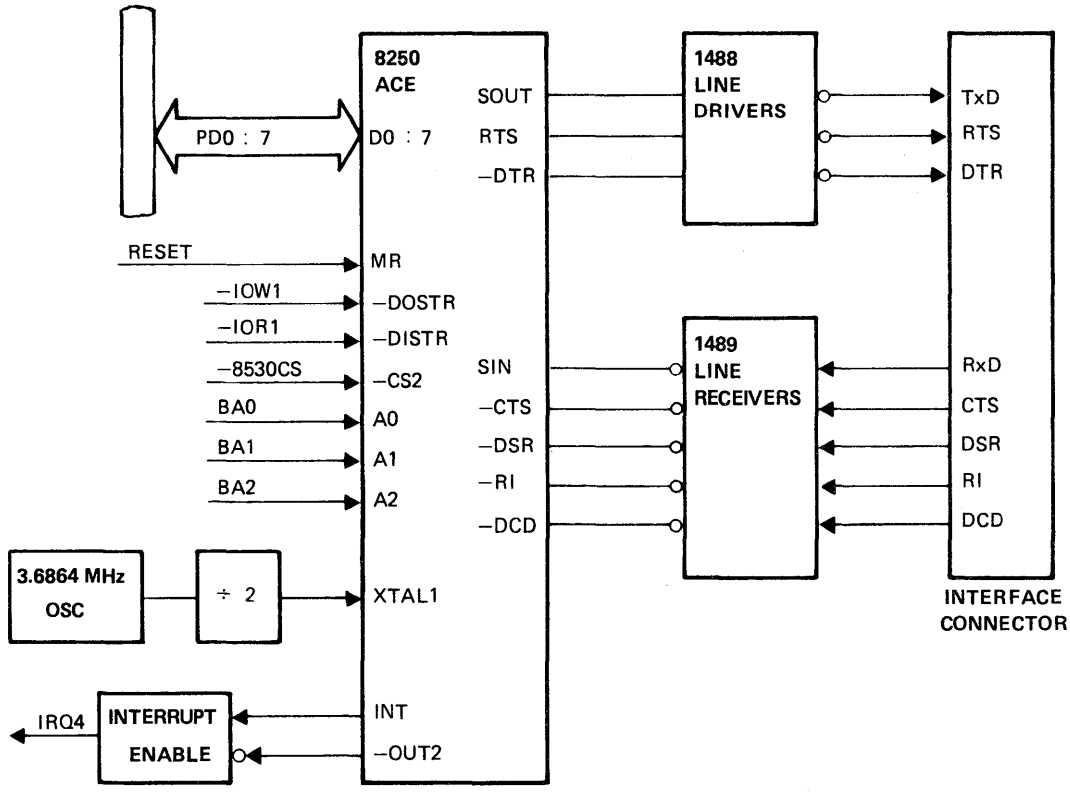


Fig. 2-36 Serial Communication Interface Block Diagram using 8250 ACE

The 8250 has a status reporting capability that includes the type and condition of the transfer being performed, as well as any error conditions such as parity error, overrun error, framing error and break interrupt. It can add or delete start bits, stop bits and parity bits to and from the serial data stream.

It also includes a programmable baud rate generator and a prioritized interrupt system that controls transmit, receive, error, line status and data set interrupts.

The different modes of operation of the 8250 are programmed by the system software that accesses or controls the contents of the 8250 registers via the peripheral data bus. Address bits BA0, BA1 and BA2 select registers to read from or write to this data bus as shown in the following table. Note that the state of the Divisor Latch access bit of the line control register affects the selection of certain registers. The DLAB must be set high by the system software to access the baud rate generator divisor latches.

DLAB	BA2	BA1	BA0	Register
0	0	0	0	Receiver Buffer (read) Transmitter Holding Register (write)
0	0	0	1	Interrupt Enable
X	0	1	0	Interrupt Identification (read only)
X	0	1	1	Line Control
X	1	0	0	Modem Control
X	1	0	1	Line Status
X	1	1	0	Modem Status
X	1	1	1	Scratchpad
1	0	0	0	Divisor Latch (LSB)
1	0	0	1	Divisor Latch (MSB)

The data format of the serial data is as follows:

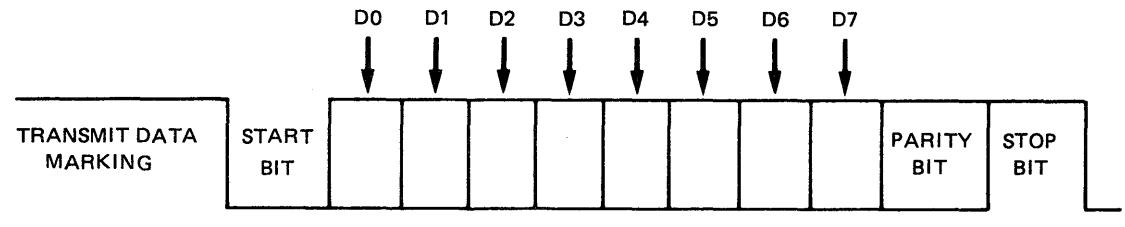


Fig. 2-37 Data Format

Data bit 0 is the first to be transmitted or received on the interface. The Serial Communications interface inserts the start bit, the correct parity bit, if programmed, and the stop bit according to the content of the line control register.

The system programmer may access or control the 8250 registers. For a description of these registers refer to the "M24/M21 Hardware Architecture and Function" manual.

Z8530 SERIAL COMMUNICATION CONTROLLER

The Z8530 SCC Serial Communication Controller used is a dual channel controller which can be configured for asynchronous as well as synchronous communication. It functions as a serial to parallel, parallel to serial controller. The two full duplex channels are independent with a transmission speed of up to 1Mbit/second, each with a separate crystal oscillator and baud rate generator.

In the asynchronous modes, transmission and reception can be accomplished independently on each channel with five to eight bits per character, plus optional even or odd parity. The transmitters can supply one, one-and-a-half, or two stop bits per character and can provide a break output at any time.

In the synchronous modes, communication can be done with internal or external character synchronization on one or two synchronous characters.

The Z8530 SCC contains thirteen write registers in each channel that are programmed by the system separately to configure the function of the channels.

Register addressing is direct for the data registers only, that are selected by a high on the Data/Control Select input D/-C. In all other cases, with the exception of Write Register 0 (WR0) and Read Register 0 (RR0), programming the write registers requires two write operations and reading the read registers requires a write and a read operation. The first write is to WR0 and contains three bits that point to the selected register. The second write is the control word for the selected register and, if the second operation is read, the selected read register is accessed. All registers, including the data registers, may be accessed in this way. The pointer bits are cleared after the read or write operation so that WR0, or RR0, is addressed again.

The system program first issues a series of commands to initialize the basic mode of operation. This is followed by other commands to qualify conditions within the selected mode. For example, the asynchronous mode, character length, clock rate, number of stop bits, even or odd parity might be set first. Then the interrupt mode would be set, and finally, receiver or transmitter enable.

For the description of these registers refer to the "M24/M21 Hardware Architecture and Function" manual.

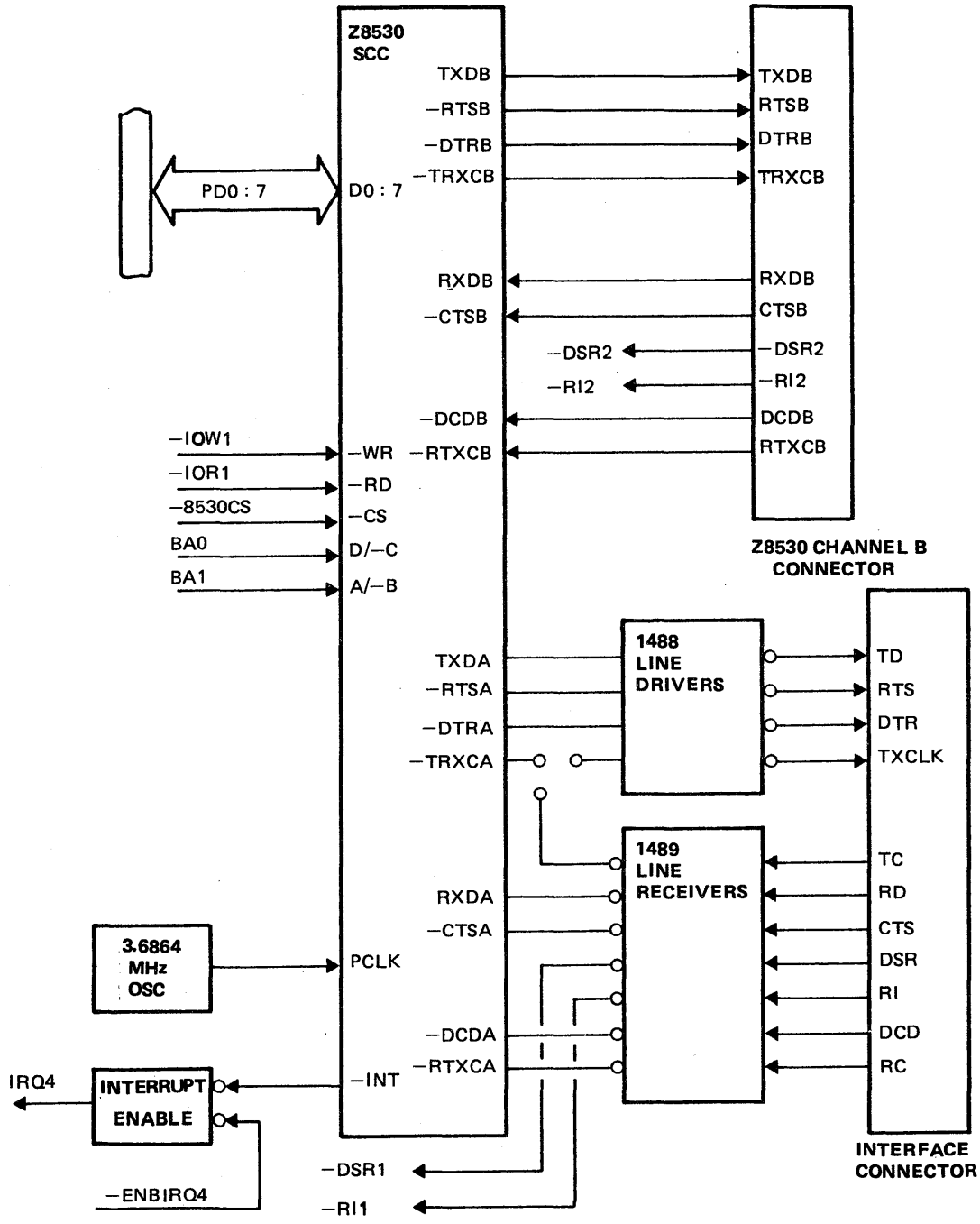


Fig. 2-38 Serial Communication Interface Block Diagram using Z8530 SCC

The above figure is a block diagram of the serial communication interface implementing the Z8530 SCC. The signal -8530CS selects the SCC while BA1 selects channel A when high and channel B when low. The signal BA1 defines the type of information transferred to or from the SCC. A high input means that data is transferred while a low means that a command is being transferred. The signals -IOW1, -IOR1 define the direction of transfer i.e. read or write to/from the SCC.

The channel A I/O lines pass through the 1488 line drivers or the 1489 line receivers and end on the interface connector. This connector is located on the motherboard and is an RS-232-C EIA standard connector as described later. The jumper, JPR-SPDT, positioned between the SCC and the line drivers/receivers, decides the direction of the transmit clock when in synchronous mode. Thus, the transmit clock can be synchronized either to the DCE (modem) clock or to the DTE (system) clock when in the synchronous modes.

The channel B I/O lines go to a 16 pin connector which is located on the motherboard. This connector allows the connection of the Mutifunction Communications Interface by means of a ribbon cable to provide the second serial communication connector which can be either an RS-232, RS-422, or a Current Loop connector.

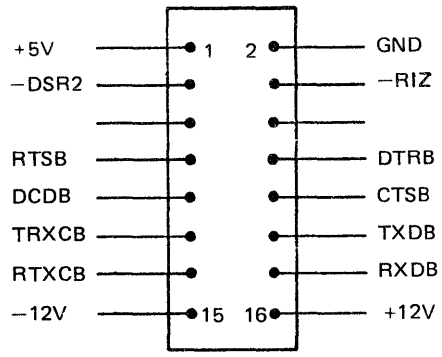


Fig. 2-39 Z8530 SCC Channel B Connector

SERIAL INTERFACE SIGNALS AND CONNECTOR

The Electronic Industries Association (EIA) recommended standard RS-232-C defines the interchange circuit signal characteristics for interfacing Data Terminal Equipment (DTE) to Data Communications Equipment (DCE).

For Data Interchange Circuits a voltage more positive than +3V with respect to ground is used to represent a logic level 0. A voltage more negative than -3V with respect to ground represents a logic level 1.

For Timing and Control Interchange Circuits a positive voltage greater than +3V with respect to ground is a logic level 1 while a voltage more negative than -3V with respect to ground is a logic level 0.

Interface Connector

The EIA RS-232-C interchange circuit signals are made available through a 25-pin, D-shell connector. This connector protrudes through the rear panel of the system box to facilitate the connection of the modem or data set cable.

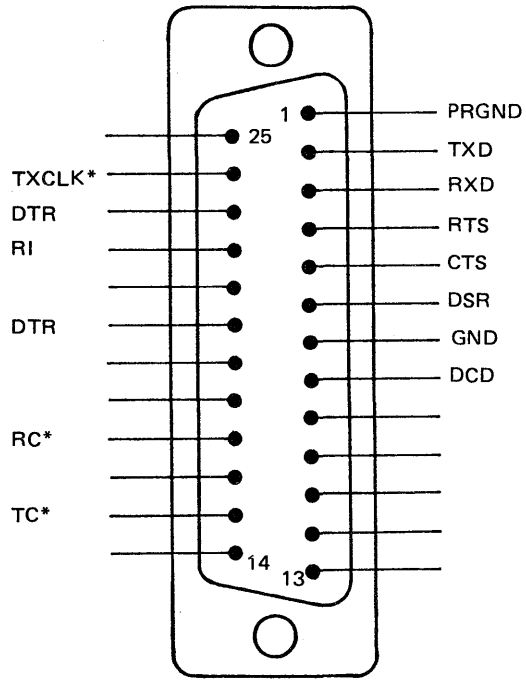


Fig. 2-40 Channel A Serial Interface Connector

The following is a description of the signals available on the channel A interface connector. I/O direction is with respect to the motherboard.

Interchange Signal	Cct	I/O	Description
PRGND	AA	-	Protective Ground - Connected to Basic Module frame.
GND	AB	-	Signal Ground/Common Return - Common ground reference for interchange circuits, except Protective Ground.
TxD	BA	O	Transmitted Data, to DCE - Generated by data terminal equipment and transferred to local modem or data set for transmission over the communication channel to the remote data terminal equipment.
RxD	BB	I	Received Data, from DCE - Generated by local modem or data set in response to data signals received over the communication channel from remote data terminal equipment .
RTS	CA	O	Request to Send, to DCE - Used to condition the local modem or data set for data transmission.
CTS	CB	I	Clear to Send, from DCE - Used to indicate whether or not the modem or data set is ready to transmit data.
DSR	CC	I	Data Set Ready, from DCE - Used to indicate the status of the local modem or data set.
DTR	CD	O	Data Terminal Ready, to DCE - Used to control the switching of modem or data set to the communication channel.
RI	CD	I	Ring Indicator, from DCE - Indicates that a ringing signal is being received on the communication channel.
DCD	CF	I	Receive Line Signal Detect (Data Carrier Detected), from DCE - Used to indicate that the data carrier has been detected by the modem or data set.

MULTIFUNCTION COMMUNICATIONS INTERFACE

The serial communications interface may be expanded to provide two communication channels. In this case an 8530 dual channel synchronous serial communication controller is used to generate the required interfacing circuit signals. The 8250 ACE is removed from its socket and the Z8530 SCC is installed in an other socket. The line receivers and drivers and connector for the second channel are located on the Multifunction Communication Interface board which must be bolted to the rear panel of the system box. Although it occupies one expansion slot it does not plug into the slot.

This board provides the line drivers for an RS-232-C channel, or an RS-422 channel or a 20mA Current Loop channel. The selection of the mode of receiving and transmitting data depends on the drivers installed on the board together with the jumper settings.

RS-232-C INTERFACE

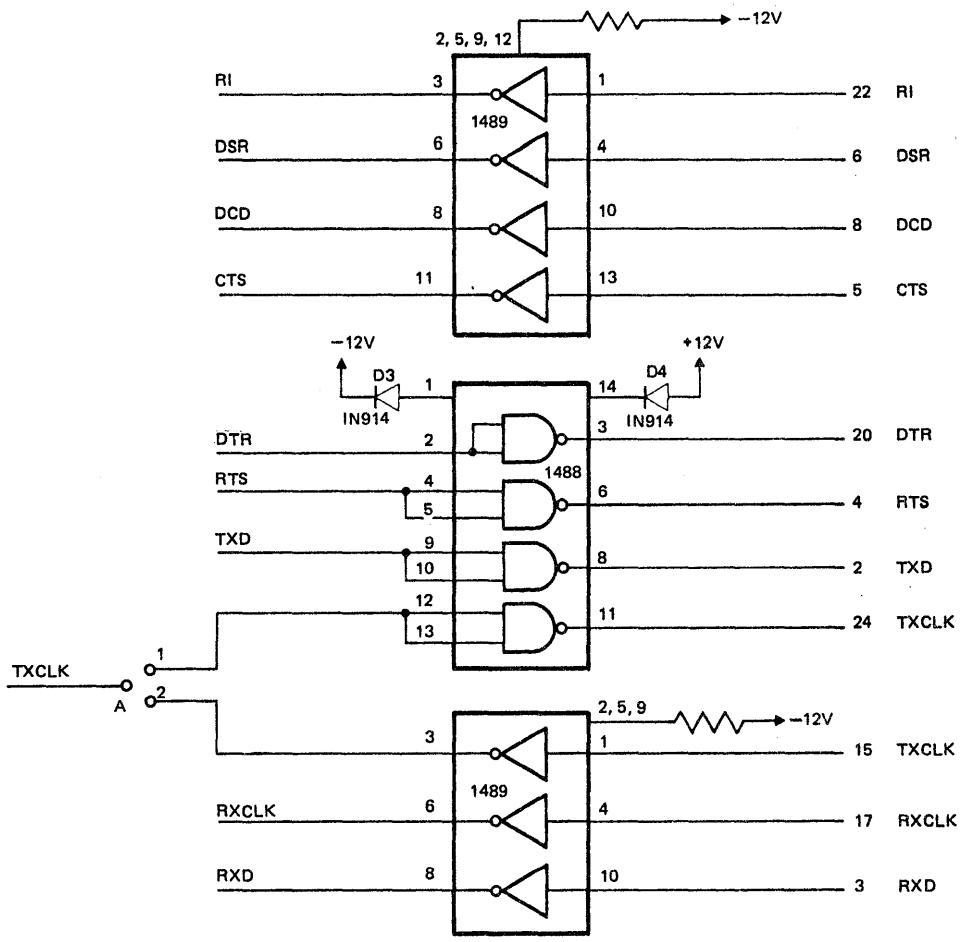


Fig. 2-41 RS-232-C Drivers

The line drivers used on this board for an RS-232-C interface are the two 1489 line receivers and the 1488 line transmitter. The 26LS31 driver must be removed from the board when using this interface.

The 1489 and 1488 line drivers converts the TTL signal levels into the RS-232-C signal levels and vice versa. The jumper A determines if the transmit clock is generated externally by the user equipment or internally by the system. The receive clock and the receive data are generated from the user equipment.

The jumper configuration for an RS232 interface is as follows:

A - 1 TRANSMIT CLOCK GENERATED INTERNALLY
A - 2 TRANSMIT CLOCK GENERATED EXTERNALLY
B - 1 RECEIVE CLOCK FROM RS232 INTERFACE
C - 1 RECEIVE DATA FROM RS232 INTERFACE

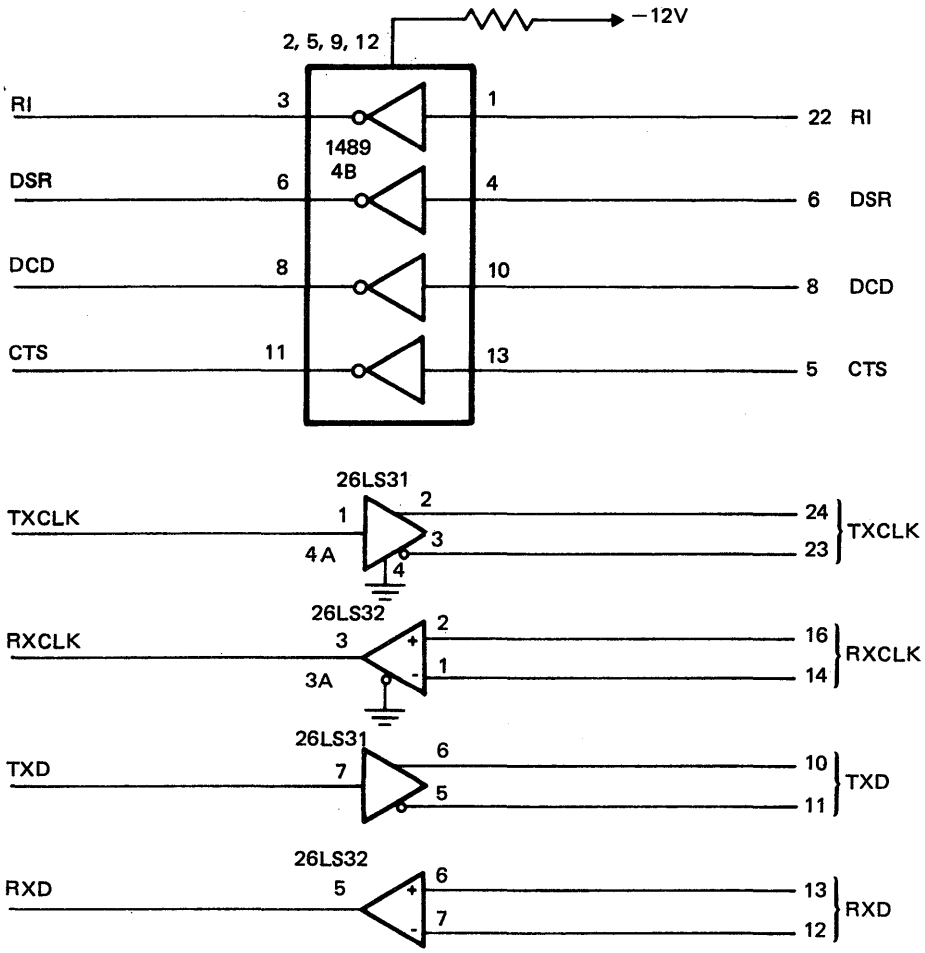


Fig. 2-42 RS422 Drivers

The difference between the RS232 interface and the RS422 interface is that in the RS232 interface the circuits are unbalanced while in the RS422 the circuits are balanced. In this case, only four circuits are balanced, namely, TxCLK, RxCLK, TxD, and RxD.

The line drivers used are 1489 for unbalanced circuits and the 26LS31 and 26LS32 for the balanced circuits. The 1488 line driver must be removed from the board when using this interface.

The jumper configuration for an RS422 interface is as follows:

A - 1 PROVIDES TRANSMIT CLOCK
B - 2 RECEIVE CLOCK FROM RS422 INTERFACE
C - 2 RECEIVE DATA FROM RS422 INTERFACE

20mA CURRENT LOOP INTERFACE

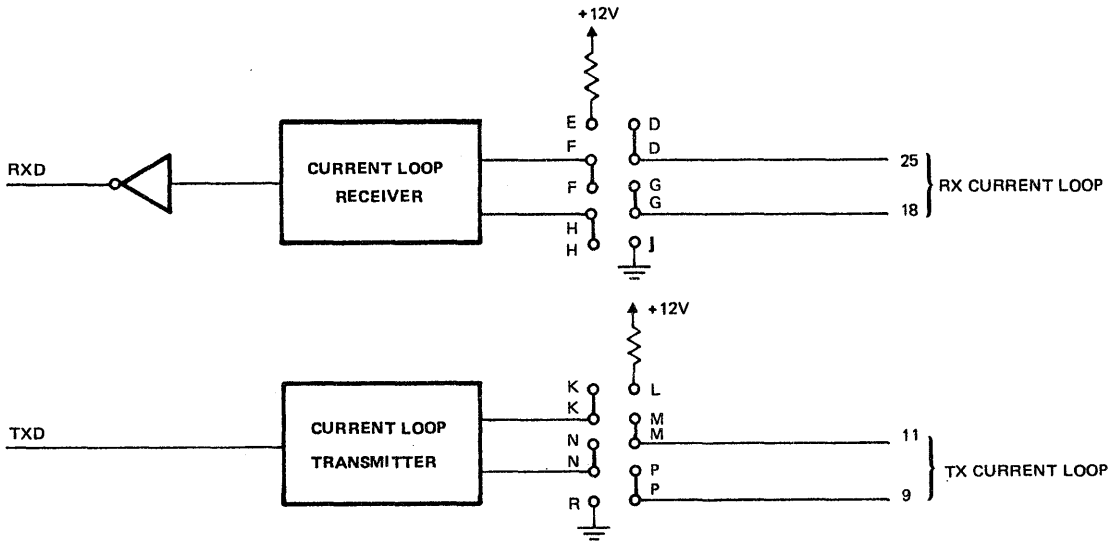


Fig. 2-43 Current Loop Drivers

The 20mA current loop interface consists of a transmitter circuitry and a receiver circuitry for modulating/demodulating of data signals coming from the Z8530 SCC. The jumpers available in this circuitry enables the selection of the current source i.e. either the system or the user equipment. The 26LS31 line driver must be removed from the board when this interface is being used.

The jumper configuration for a 20mA current loop interface is as follows:

TRANSMIT ACTIVE
K - L
N - M
R - P

TRANSMIT PASSIVE
K - M
N - P

RECEIVE ACTIVE
D - E
G - F
J - H

RECEIVE PASSIVE
D - F
G - H

3. BUS CONVERTER BOARD

CONTENTS

INTRODUCTION

The purpose of the Bus Converter Board is to provide 16 bit and 8 bit system expansion slots that accept and allow the simultaneous use of Olivetti and IBM compatible expansion boards.

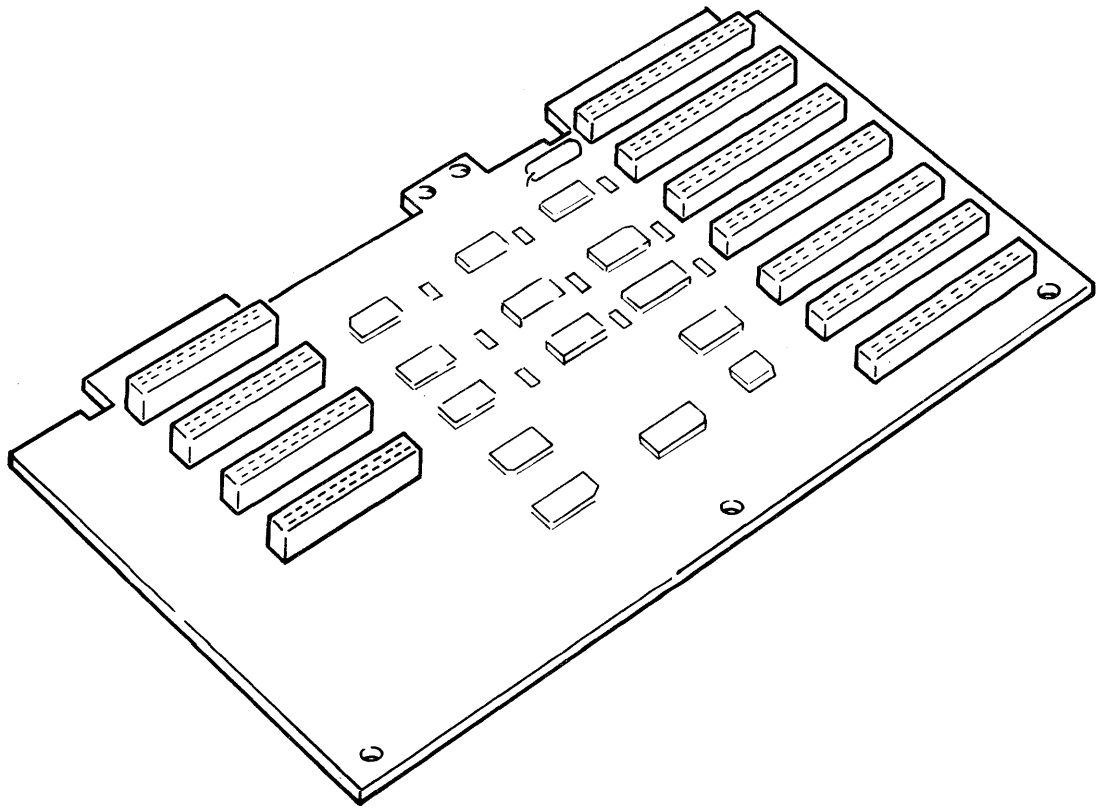


Fig. 3-1 M24 Bus Converter Board

The Bus Converter Board provides two sets of I/O bus connectors:

- Four 38 pin connectors to handle signals used by the 16 bit Olivetti boards (APB Z8000 Board, etc).

- Seven 62 pin connectors to handle signals used by both the 16 bit Olivetti boards and the 8 bit IBM compatible boards (SDLC, IEEE488 etc).

The Bus Converter Board plugs into the display controller board, a plane above the motherboard. The M21 Bus Converter Board circuitry is the same as the M24 Bus Converter Board but it only contains three 62 pin connectors and one 38 pin connector.

16 BIT DATA TRANSFERS

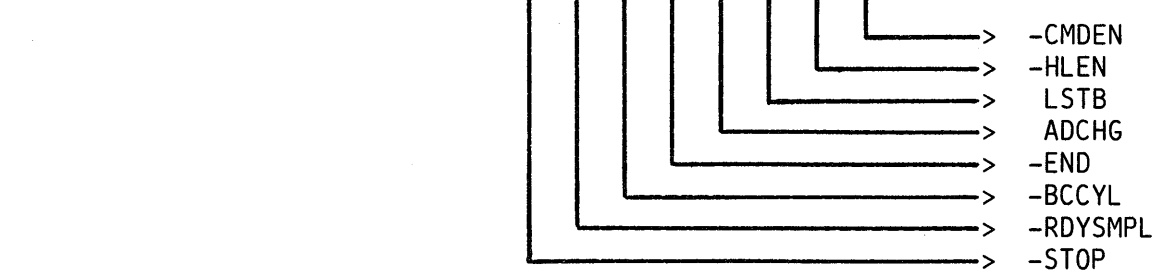
During 16 bit data transfers, expansion boards dialogue directly with the motherboard 16 bit data bus.

The signal -16BCH determines whether an 8 bit bus or a 16 bit bus is being addressed. It is generated by all the Olivetti boards including motherboard, memory expansion board, display controller board, and I/O expansion boards.

When a 16 bit transfer occurs, signal -16BCH goes low pulling the signal CYCLE low with it. The counter 74LS161 is then loaded with four zeroes (as inputs A, B,C,D are tied to ground). The outputs of this counter (in this case all zeroes) are used as address inputs to the 32 x 8 bipolar PROM 82S123. When all the address bits are zero the outputs of the PROM and the bus converter board signals are set according to the following table:

PROM Address **PROM Output**

Hex	A4 A3 A2 A1 A0	Hex	B7 B6 B5 B4 B3 B2 B1 B0
x0	x 0 0 0 0	F3	1 1 1 1 0 0 1 1



Signal -CMDEN is input to the 4 OR gates and disables the IBM compatible read and write command lines -IMRD, -IIOR, -IMWR, -IIOW.

Signal -HLEN is input to 2 OR gates in series and the final high output disables the 74LS244 low/high byte read data latch.

Signal LSTB disables the 74LS373 disables the high/low byte write data buffer.

ADCHG is inverted to produce signal -MBDIS high and this goes on to the motherboard to disable the CPU Data Buffer. Infact this buffer is only used during DMA and 8 bit transfers.

-END is NAND gated with CYCLE to produce the ready signal ORDY. This signal is input to the motherboard wait logic and when active high indicates that a 16 bit transfer can take place. When pulled low by an expansion board this signal lengthen the read or write cycles.

-RDYSMPL and -STOP are used to inhibit the counter during 8 bit data transfers.

8 BIT DATA TRANSFERS

During 8 bit data transfers, I/O expansion boards dialogue with the motherboard 16 bit data bus via the bus converter board.

If an 8 bit transfer occurs, -16BCH remains high. One of the signals -XMRD (memory read), -XIOR (input/output read), -XMWR (memory read), -XIOW (input/output write) goes active low and thus the final outcome is that signal CYCLE goes high. A low to high transition of this signal CYCLE starts the counter 74LS161. Naturally, when this occurs the outputs of the counter start changing at a rate of 8MHz until the counter is stopped. The counter stops when one of the enable inputs go low.

The outputs of the counter address inputs A0 to A3 of the PROM, while BYTE/-WORD addresses input A4 of the PROM. Address input A4 is set according to whether the data transfer is for a byte, single cycle, or a word, double cycle, operation. Byte or word operation is determined by decoding -BHE and A0 to produce BYTE/-WORD in accordance with the following table:

-BHE	A0	BYTE/-WORD	
0	0	0	Whole Word (double cycle)
0	1	0	Higher byte (single cycle)
1	0	1	Lower byte (single cycle)
1	1	1	none

As the counter counts up from zero, the outputs of the PROM and the bus converter board control signals are set in accordance with the following table:

PROM Address						PROM Output								
Hex	**A4**	**A3**	**A2**	**A1**	**A0**	**Hex**	**B7**	**B6**	**B5**	**B4**	**B3**	**B2**	**B1**	**B0**	**Count**
	Double Cycle														
00	0	0	0	0	0	F3	1	1	1	1	0	0	1	1	0
01	0	0	0	0	1	F3	1	1	1	1	0	0	1	1	1
02	0	0	0	1	0	F3	1	1	1	1	0	0	1	1	2
03	0	0	0	1	1	DB	1	1	0	1	1	0	1	1	3
04	0	0	1	0	0	D8	1	1	0	1	1	0	0	0	4
05	0	0	1	0	1	DC	1	1	0	1	1	1	0	0	5
06	0	0	1	1	0	9C	1	0	0	1	1	1	0	0	6
07	0	0	1	1	1	DC	1	1	0	1	1	1	0	0	7
08	0	1	0	0	0	D8	1	1	0	1	1	0	0	0	8
09	0	1	0	0	1	D9	1	1	0	1	1	0	0	1	9
0A	0	1	0	1	0	DB	1	1	0	1	1	0	1	1	10
0B	0	1	0	1	1	D3	1	1	0	1	0	0	1	1	11
0C	0	1	1	0	0	D2	1	1	0	1	0	0	1	0	12
0D	0	1	1	0	1	D2	1	1	0	1	0	0	1	0	13
0E	0	1	1	1	0	92	1	0	0	1	0	0	1	0	14
0F	0	1	1	1	1	42	0	1	0	0	0	0	1	0	15
	Single Cycle														
10	1	0	0	0	0	F3	1	1	1	1	0	0	1	1	0
11	1	0	0	0	1	F3	1	1	1	1	0	0	1	1	1
12	1	0	0	1	0	F3	1	1	1	1	0	0	1	1	2
13	1	0	0	1	1	D1	1	1	0	1	0	0	0	1	3
14	1	0	1	0	0	D0	1	1	0	1	0	0	0	0	4
15	1	0	1	0	1	D4	1	1	0	1	0	1	0	0	5
16	1	0	1	1	0	94	1	0	0	1	0	1	0	0	6
17	1	0	1	1	1	D4	1	1	0	1	0	1	0	0	7
18	1	1	0	0	0	44	0	1	0	0	0	1	0	0	8

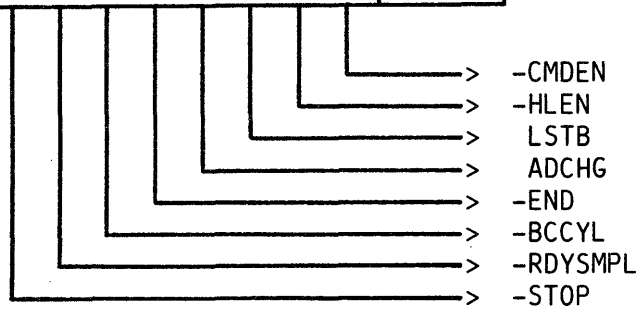


Figure 3-2 shows the timing diagram of the PROM outputs during an 8 bit byte cycle. Figure 3-3 shows the timing diagram for an 8 bit word cycle. The difference between these cycles is that the input A4 to the PROM is low on a word operation (A0 and -BHE both low) and high on a byte operation.

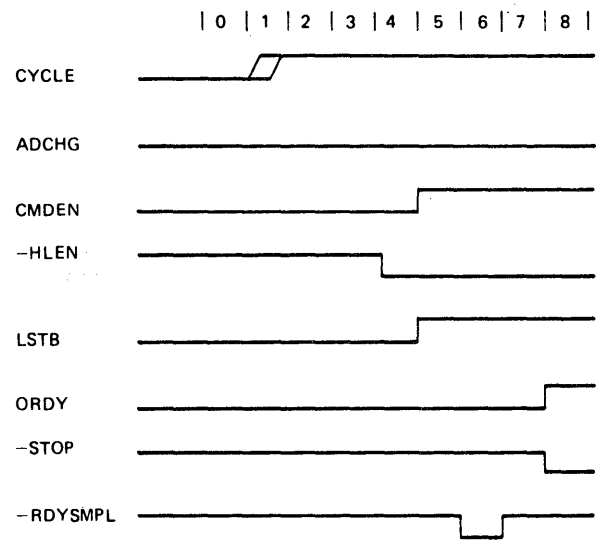


Fig. 3-2 PROM Outputs Timing for Byte Cycle

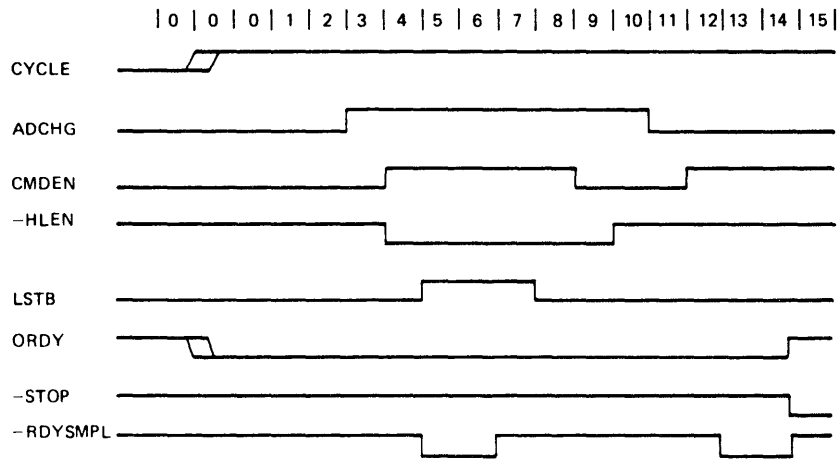


Fig. 3-3 PROM Outputs Timing for Word Cycle

When both inputs to the NAND gate (CYCLE and -END) are high, ORDY is low and this activates the wait logic circuitry on the motherboard.

When -CMDEN goes low it enables the 4 OR gates 74LS32 and permits any of the signals -XMRD, -XIOR, -XMWR, -XIOW to go through the OR gates and activate one of the signals -IMRD, -IIOR, -IMWR, -IIOW depending on the operation to be performed on the 8 bit bus board.

When ADCHG goes high (for byte operations it is always low) it gets inverted to become -MBDIS which in turn is input to the motherboard logic to enable the CPU data buffer. ADCHG is also input to the exclusive OR gate 74LS86 and on a word operation drives the signal IA0 high.

When -HLEN goes low it enables the 74LS244 latch if a write operation is in progress. Thus data bits D8-D15 go through to the 8 bit board.

When LSTB goes high it enables latch 74LS373. Hence data bits D0-D7 pass through this latch when pin 1 of this latch goes low. This happens on a read operation and when signal -BHE is low. Thus data bits D0-D7 pass through this latch to become D8-D15. When -BHE is high, the data lines D0-D7 go directly to the motherboard as the lower byte.

When PROM output -RDYSMPL goes active low, it is input to an OR gate 74LS32. The other OR input is the 1RDY signal from the 8 bit bus board. If this signal 1RDY is low at the time -RDYSMPL is low, the counter is stopped in order to accomodate for the lengthy data transfer.

When PROM output -STOP goes low, it stops the counter.

When the latch output -END goes low, it drives the NAND gate output ORDY high disactivating the wait logic on the motherboard.

This board also generates the 14 MHZ clock needed by some I/O boards. -5 Volts are also output from this board.

I/O EXPANSION BUS CONNECTORS

Figure 3-4 shows the two types of connector slots with their respective signal names.

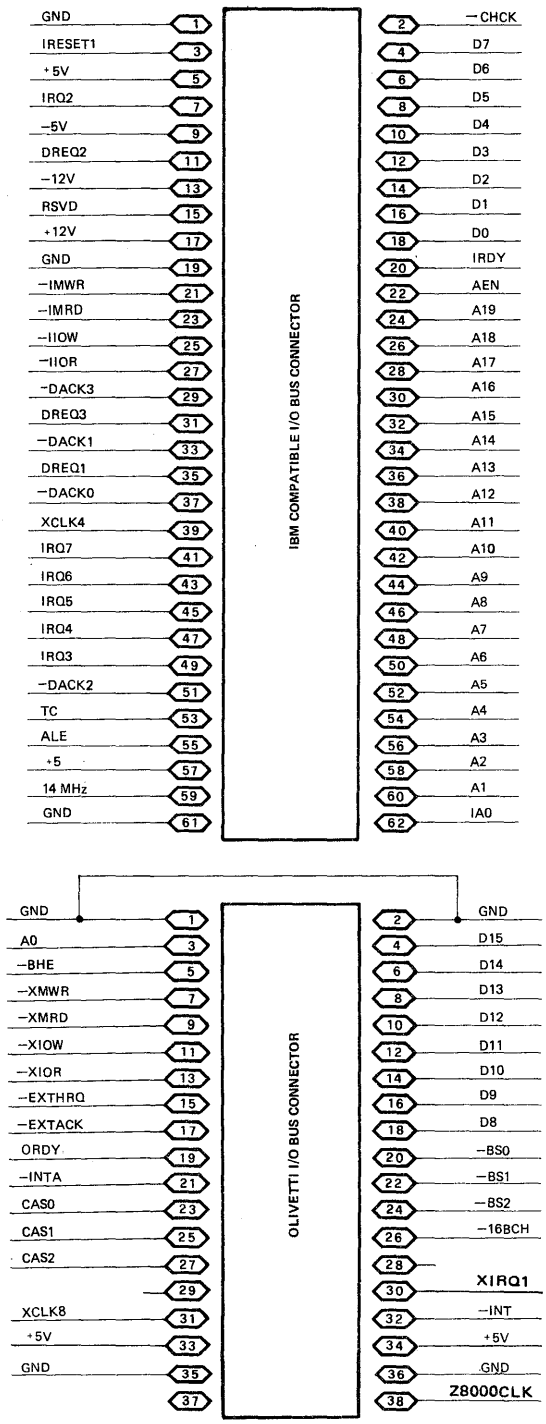


Fig. 3-4 I/O Expansion Bus Connectors

IBM COMPATIBLE I/O EXPANSION BUS SIGNALS

The following is a description of the signals on the IBM compatible I/O expansion bus.

Signal	I/O	Description
14MHz	O	14.31818 MHz Clock - High-speed clock with a 70ns period and a 50% duty cycle.
XCLK4	O	4 MHz Clock - Divide-by-two of CPU clock with a 200ns period and a 50% duty cycle.
IRESET1	O	Reset - This line is used to reset or initialize system logic on power-on or hardware reset. This line is active high.
IA0, A1 to A19	O I/O	Address Bits - These lines are used to address memory and I/O devices in the system. The 20 address lines allow access of up to 1 Mbyte of memory. These line are generated by the CPU, NDP or the DMA controller and are active high.
D0 to D7	I/O	Data Bits - These lines provide data bus bits for the CPU, NDP, memory, and I/O devices. These lines are active high.
ALE	O	Address Latch Enable - This line is available on I/O Bus as an indicator of a valid address (when used with AEN). Addresses are latched on the falling edge of ALE.
-CHCK	I	Channel Check - This line is used for reporting error conditions from I/O devices. Memory expansion options use this line for reporting parity errors. An error is indicated when this line is active low.
IRDY/BCRDY	I	IBM Ready - This line, normally high (ready), is pulled low (not ready) by a memory or I/O device to lengthen I/O or memory cycles. It allows slow devices to be connected to the I/O Bus with a minimum of difficulty. Any slow device using this line should drive it low immediately upon detecting a valid address and a read or write command. This line should not be low longer than 10 clock cycles. Machine cycles (memory or I/O are lengthened by an integral number of 4MHz cycles (200ns).

Signal	I/O	Description
IRQ2 to IRQ7	I	Interrupt Request - These lines are used to signal interrupt controller that an I/O device requires attention. An interrupt request is generated by raising an IRQ line (low to high) and holding it high until it is acknowledged by the interrupt service routine. IRQ4 is generated by the motherboard serial interface, IRQ6 is generated by the motherboard diskette drive interface and IRQ7 is generated by the motherboard parallel interface
-IIOR	O	I/O Read Command - This line instructs an I/O device to drive its data onto the data bus. This line may be driven by the CPU or the DMA Controller and is active low.
-IIOW	O	I/O Write Command - This line instructs an I/O device to read the data on the data bus. This line may be driven by the CPU or the DMA Controller and is active low.
-IMRD	O	Memory Read Command - This line instructs the memory to drive its data onto the data bus. This line may be driven by the CPU or the DMA Controller and is active low.
-IMWR	O	Memory Write Command - This line instructs the memory to read and store the data present on the data bus. This line may be driven by the CPU or the DMA Controller and is active low.
DREQ1 to DREQ3	I	DMA Request - These lines are asynchronous channel requests used by peripheral devices to gain DMA service. They are prioritized with DREQ4 having the lowest priority and DREQ1 the highest. A request is generated by taking a DREQ line active high. A DREQ line must be held high until the corresponding DACK line goes active. DREQ2 is used by the Diskette Drive Interface on the motherboard.
-DACK0 to -DACK3	O	DMA Acknowledge - These lines are used to acknowledge DMA requests, lines DACK1 to DACK3, and to refresh dynamic memory, DACK0. These lines are active low.
AEN	O	Address Enable - This line indicates that the CPU has released control of the system busses to allow DMA transfers to take place. When this line is active high, the DMA controller has control of the address bus, data bus, memory and I/O read command lines, and memory and I/O write command lines.
TC	O	Terminal Count - This line provides a pulse when the terminal count for any DMA channel is reached. This line is active high.

Signal	I/O	Description
+5V	-	+5V dc - Located on 2 contacts.
-5V	-	-5V dc - Located on 1 contact.
+12V	-	+12V dc - Located on 1 contact.
-12V	-	-12V dc - Located on 1 contact.
GND	-	Ground - Located on 3 contacts.

OLIVETTI I/O EXPANSION BUS SIGNALS

The following is a description of the Olivetti I/O expansion bus signals not common to those on the IBM compatible I/O expansion bus.

Signal	I/O	Description
XCLK8	O	8 MHz Clock - CPU clock with a 100ns period and a 33% duty cycle.
A0	O	Address Bit - This address line is used in conjunction with -BHE to enable data onto the low byte D0 to D7.
-BHE	O	Bus High Enable - This line is used in conjunction with A0 to enable data onto the high byte D8 to D15.
D8 to D15	I/O	Data Bits - These lines provide the high order byte for 16 bit wide data transfers.
XIRQ1	I/O	Interrupt Request - This line is derived from IRQ1 generated by the keyboard controller on the motherboard and made available for general use.
-XIOR	O	I/O Read Command - This line instructs an I/O device to drive its data onto the data bus. This line may be driven by the CPU or the DMA Controller and is active low.
-XIOW	O	I/O Write Command - This line instructs an I/O device to read the data on the data bus. This line may be driven by the CPU or the DMA Controller and is active low.
-XMRD	O	Memory Read Command - This line instructs the memory to drive its data onto the data bus. This line may be driven by the CPU or the DMA Controller and is active low.
-XMWR	O	Memory Write Command - This line instructs the memory to read and store the data present on the data bus. This line may be driven by the CPU or the DMA Controller and is active low.

Signal	I/O	Description
EXTHRQ	I	External Hold Request - This line indicates to the Bus Arbiter on the motherboard that an external processor is requesting control of the system busses.
-EXTACK	O	External Acknowledge - This line is used to acknowledge external hold requests and indicate to the external processor that the CPU has released control of the system busses. This line is active low.
-BS0 to -BS2	O	Status Bits - These lines are used to provide CPU status information.
ORDY	I	Olivetti Ready - This line, normally high (ready), is pulled low (not ready), by a memory or I/O expansion board to lengthen I/O or memory cycles.
INTA	O	Interrupt Acknowledge - This line is used by the CPU to acknowledge the interrupt request from the 8259A interrupt controller on the motherboard. This line is active low.
CAS0 to CAS2	O	Cascade Lines - These lines are used to expand vector interrupt levels. They are outputs from the master 8259A interrupt controller to slave interrupt controllers.
-16BCH	I	16 Bit Channel - This line indicates that a 16 bit I/O Expansion Board is being addressed. This line is active low.

4. MEMORY EXPANSION BOARD

CONTENTS

INTRODUCTION

The memory expansion board is organized in three banks of 128KB each. The three banks, banks 0, 1, and 2, are each made up of 2 groups of nine 64K x 1 RAM chips - the lower byte (even addresses) and the higher byte (odd addresses). The ninth RAM chip is used for parity checking.

This board is inserted into one of the slots of the Bus Converter board of the M24/M21 system. One has the option of using all three banks, two banks or just one bank. The minimum memory storage is 128KB (one bank filled with 64K x 1 chips) and the maximum storage 384KB (all three banks filled up).

MEMORY CONTROL LOGIC

Like the motherboard, the expansion memory can be addressed either as bytes or as words (2 bytes). The signals required to address and access the expansion memory are:

RAS0	row address strobe for bank 0
RAS1	row address strobe for bank 1
RAS2	row address strobe for bank 2
CASL0	column address strobe for lower byte (even) bank 0
CASH0	column address strobe for higher byte (odd) bank 0
CASL1	column address strobe for lower byte (even) bank 1
CASH1	column address strobe for higher byte (odd) bank 1
CASL2	column address strobe for lower byte (even) bank 2
CASH2	column address strobe for higher byte (odd) bank 2
MA0-7	memory address lines
MD0-15	memory data lines
XMWR1	memory write for bank 0
XMWR2	memory write for bank 1
XMWR3	memory write for bank 2
PALRD	parity lower read (even)
PAHRD	parity higher read (odd)
PALWR	parity lower write (even)
PAHWR	parity higher write (odd)

When the 8086 CPU status lines indicate that a read or write is to be performed, the signals -BS0, -BS1, -BS2 (outputs of LS244 latch on motherboard) are input to the circuitry on the expansion memory that activates signal LDRAM. LDRAM is active when signals -BS0--BS2 are set according to the following table:

-BS2	-BS1	-BS0	
1	0	0	Read Memory
1	0	1	Read Memory
1	1	0	Write Memory

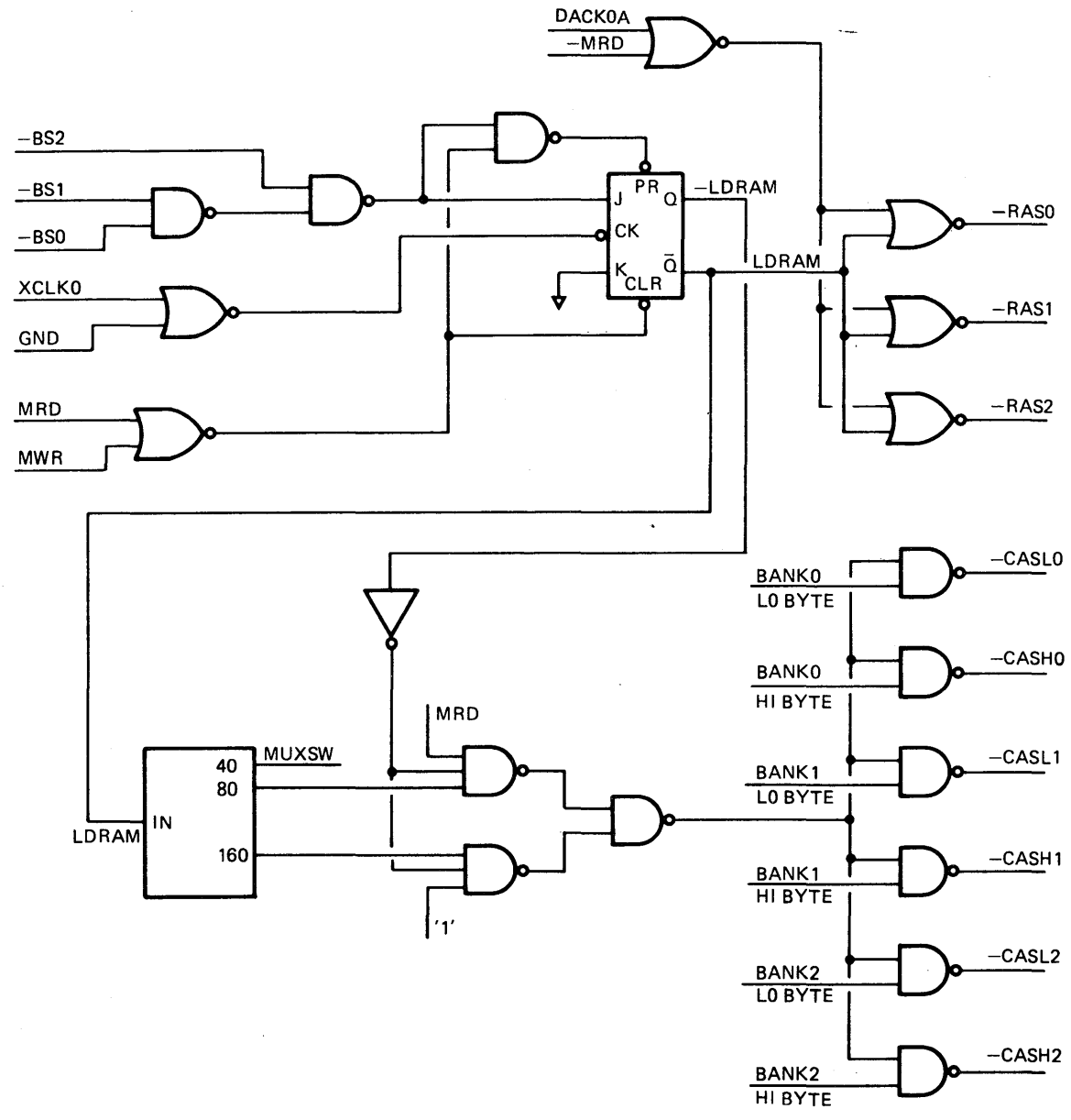


Fig. 4-1 Memory Control Logic

With LDRAM active, all the row address strobes -RAS0 to -RAS2 are active low.

Decoder 74LS138 decodes whether the memory address is in bank 0, 1 or 2. The following table illustrates the truth table.

A19	A18	A17	-BANK0	-BANK1	-BANK2	
0	1	0	0	1	1	BANK 0
0	1	1	1	0	1	BANK 1
1	0	0	1	1	0	BANK 2

Signals A0 and -BHE coming from the motherboard determine which byte (higher or lower) is going to be selected. On a word operation both bytes are selected.

-BHE	A0	
0	0	Word (2 bytes)
0	1	High byte (odd address)
1	0	Low Byte (even address)

Having selected the correct byte and the desired bank, the corresponding CAS signal is activated. For a memory read, the 80ns tap output of the delay line generates the desired CAS signal. For a memory write the 160ns tap output of the delay line generates the desired CAS signal.

MEMORY ADDRESSING

The addresses for the expansion memory DRAM travel on the A bus. A1-A7, A9-A14, and A16 go to the RAM address multiplexer which consist of two 2 to 1 74S158 multiplexers.

When the address bits are input to the multiplexers, the select input pin 1 signal MUXSW is low. Thus the bits A1-A7 are output from the multiplexer as MA0-MA5 and MA7 respectively. MA6 is generated by the circuitry shown in fig. 4-2. MA6 follows the inverse of A8 when MUXSW signal is low and the inverse of A15 if MUXSW is high as long as -DACK0A is not active low.

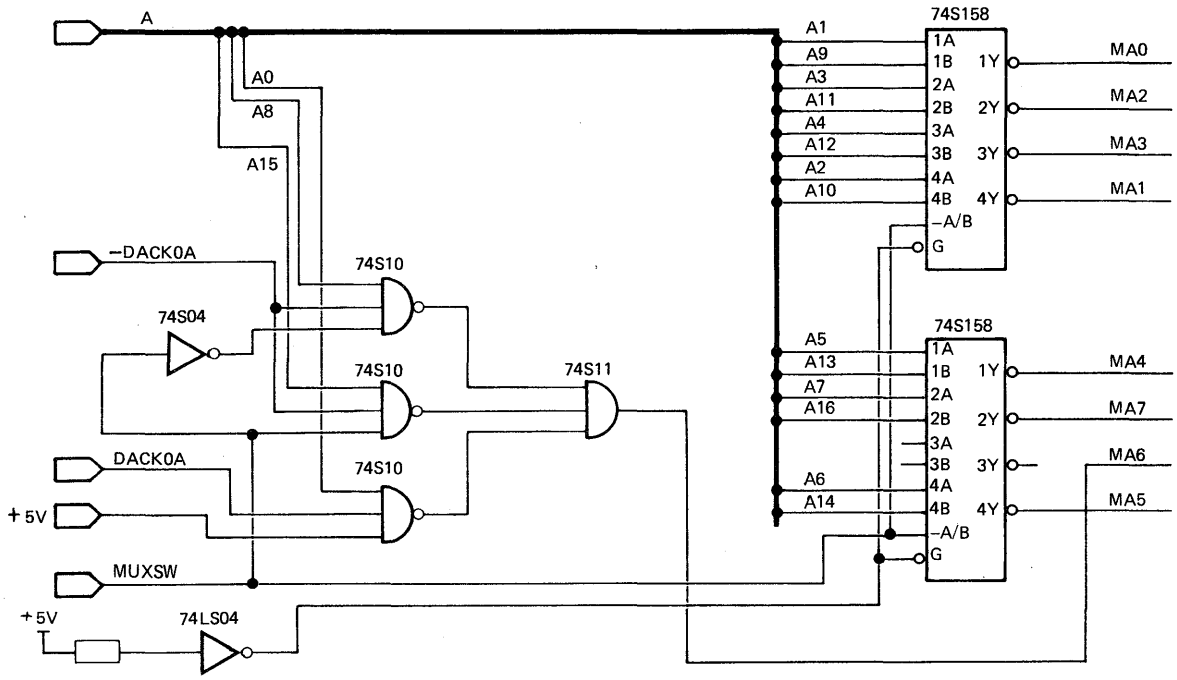


Fig. 4-2 Address Multiplexers

Thus when the -RAS signals go active low, the outputs of the multiplexer, MA0-MA5 and MA7, and MA6 are latched into the RAMs as the row address for all the banks.

40ns after -RAS0, -RAS1 and -RAS2 go active, MUXSW signal goes high, and the address bits A9-A16 generate the signals MA0-MA7. When -CASH or -CASL or both signals for the desired bank go active, MA0-MA7 are latched into the desired RAM bank as the column address for either the even or odd byte (or both in the case of word addressing). In a read cycle the relevant CAS signal becomes active 80ns after RAS signal, while in a write cycle it becomes active after 160ns.

Data from the addressed RAMs passes through 16 bit memory bidirectional data buffers (74LS245). These data buffers interface the MD bus to the main D bus. Odd addressed data travels on lines MD8-MD15 while even addressed data travels on lines MD0-MD7. On a word operation (16 bits to be accessed) two possibilities exist:

- First byte on even address

- First byte on odd address

If the first byte is on an even address only one memory cycle is needed for accessing two bytes. If the first byte is on an odd address two cycles are required for a word operation. For a DMA memory cycle one wait state must be inserted as was explained for the motherboard memory.

The ninth RAM chip on each byte is used for parity checking. Two parity bits are generated by the two parity generators 74S280 during a write operation and, then, these bits are read and checked during a read operation. If a memory parity error exits the signal -CHCK goes active and a Non-Maskable Interrupt is generated by the system logic.

The signals -16BCH, -BDSEL, which are generated on this board, are active low whenever one of the banks is addressed.

5. DISPLAY CONTROLLER

CONTENTS

CHARACTERISTICS

The M24/M21 display controller is designed to interface between the motherboard and the monochrome or colour displays. It attaches itself perpendicularly to the motherboard by the two lower connectors and provides another two connectors for the connection of the bus converter board.

The controller has two basic modes of operation: alphanumeric (A/N) and all points addressable graphics (APA), with additional modes available within these. In the alphanumeric mode, the display can be operated either in a 40 column by 25 row mode, or in an 80 column by 25 row mode. In both modes, character cell size is 8 dots by 16 lines.

The IBM character set is provided to have IBM software compatibility. This requires 4K bytes of ROM. When 8K ROM is present a second character set is implemented to provide for international requirements. Switching between one character set and another is done by software.

In the graphics mode four resolutions are available:

- 640 x 400 monochrome graphics
- 640 x 200 IBM compatible monochrome graphics
- 320 x 200 IBM compatible colour graphics
- 512 x 256 M20 compatible monochrome graphics (with APB Z8000 board present)

Note that monochrome, in this case, means any one colour for the foreground on a black background.

The character attributes of Reverse Video, Blinking, Highlight, Hide, and Underline are available in the alphanumeric modes. In the colour mode, 4 colours chosen from a palette of 16 can be displayed simultaneously. In the case of a monochrome monitor, instead of different colours, various shades of grey are displayed.

The Display Controller contains 32KB of arbitrated RAM for use as refresh memory, and supports one graphics bit plane.

The Display Controller is implemented using a 6845 CRT Controller device, and operates in a non-interlacing scanning mode at either 24MHz, 12MHz or 19MHz pixel clock depending on the mode and resolution selected.

DISPLAY OPTION BOARD

This display controller can be upgraded by the insertion of an option board in the bus converter expansion slots. This display option board is connected to the indigenous display controller board by means of a flat cable.
This upgrading adds the following features:

- up to 3 additional 640 by 400 bit planes.
- software controlled look up table.
- high resolution (16 by 16) characters.
- hardware smooth scroll.
- ability to display characters and graphics simultaneously.
- 4 colour, 8 colour PCOS compatibility.

This means that with the addition of the optional board, 8 and 16 colour modes are provided and so up to 16 colours can be displayed simultaneously in graphics.

PRINCIPLES OF OPERATION

There are two common ways in which a personal computer can send information to its video display screen. It can treat the display screen as it treats any other input output device, i.e. sending various commands to the display controller.

The second way of communicating with a computer display screen is called memory mapping. In memory mapping the computer writes information into the controller memory, and the screen, which is continually scanning the memory, shows whatever is in it. The CPU can also see what is on the display screen by simply reading what is in memory.

In this display controller both ways of communication are used. Memory mapping is used for displaying data while I/O commands are used to control certain aspects of the display like cursor positioning, cursor size, clearing and resetting the display, and changing modes of operation.

The display controller has its own built-in memory so that it does not use up any of the motherboard memory. This built in display memory has an arbitration circuitry so that both the CPU and the display can access the display memory rapidly without much interference. The 80 by 25 alphanumeric mode circuitry is not dual ported and updates occur only during horizontal or vertical retrace.

MEMORY MAPPING

Alphanumeric Mode

The display memory is addressed by the CPU as if it was part of the system memory. The starting address for the 1st bit plane is B8000 hex and the RAM consists of 16K x 16 words or 32K bytes.

For each screen position, there are two bytes in memory. The first byte, at an even address location, specifies the character to be displayed on the screen and contains the ASCII code for the character.

The second byte, at an odd location, is called the attribute byte. This byte controls the colour in the case of a colour display, or the shades of grey in the case of a monochrome display. It also controls blinking, brightness and underline for both kinds of display.

The first position on the screen at the top left corner uses the first two bytes in the display memory. The next position, one column to the right, uses the next two memory locations. This continues up to the end of the first row of the screen. At the end of each row, the memory map immediately goes to the beginning of the next row; so that the first column of the next row is mapped to the pair of bytes following the last column of the preceding row. There is no gap in the use of memory, and no boundary where one row ends and the next begins. This means that in switching from an 80 column mode to a 40 column mode, or vice versa, the map of rows and columns changes.

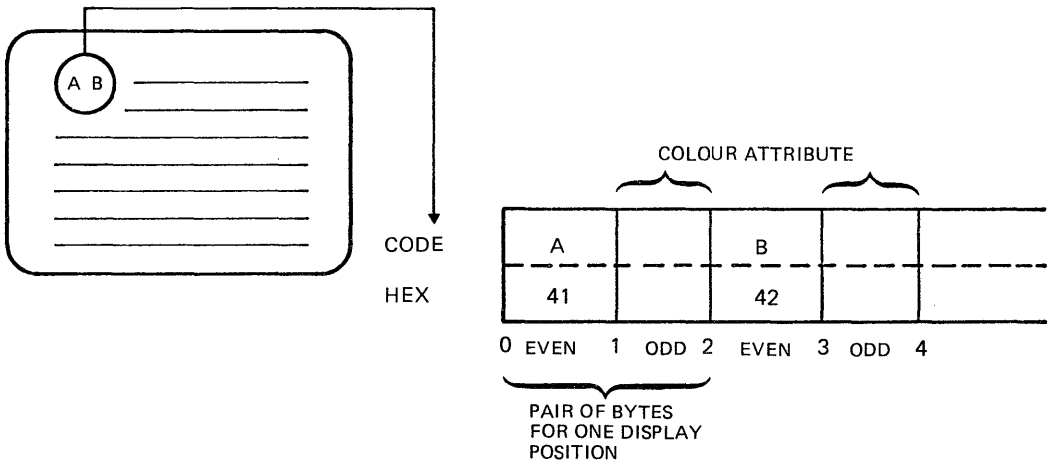


Fig. 5-1 Display Memory Map, Character Mode

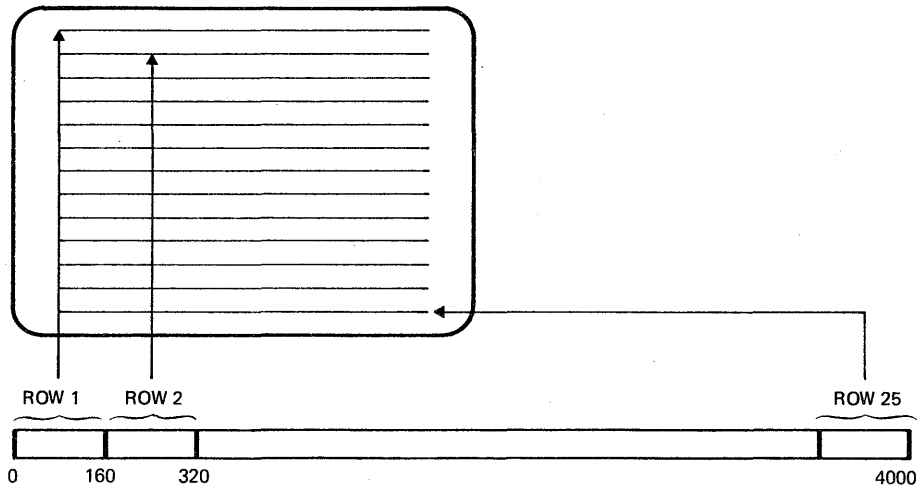


Fig. 5-2 Display Rows to Memory Locations

Thus when in 80 column mode, the display takes up 4000 bytes while when in 40 column mode only 2000 bytes are needed.

The rest of the 32K bytes of display memory is used to provide the text mode with multiple screen images called pages. So there are either eight or sixteen pages for the two text modes. At any one time, the information stored in one of the pages appears on the screen while the other pages wait for the cue for them to appear. The display controller switches from one page to another when the command is given and the screen changes immediately. The starting address for each page is at 4K intervals for the 80 column mode and at 2K intervals for the 40 column mode.

Graphics Mode

This Display Controller, when in the graphics mode, uses pixel graphics. In pixel graphics the display screen is divided up into a rectangular grid of many small picture elements called pixels. Each pixel can be lit or not lit, so that a pixel graphics display is made up of a series of small dots. Memory mapping is again used to control which pixels are to be lit and their colour.

The horizontal dimension varies according to the resolution used. This depends on the mode selected.
In medium resolution graphics, there are 320 pixels across and 200 down, while in high resolution graphics there are 640 across and 200 down or 640 across and 400 down.

In the high resolution mode, each pixel uses only one bit in memory to determine if it is lit or not. Thus, there are 640 x 200, or 128,000 pixels and to control them all, with one bit for each pixel, 128,000 bits are required. This means that 16K Bytes of RAM are used in this mode.

In the medium resolution, 320 x 200 mode, each pixel needs two bits in memory. These two bits can specify four different values which determine which of the four different colours that the pixel will have. In this mode, 16K bytes of RAM are also required.

So, for the colour display, four colours can be used in graphics mode. Three of those four will either be from the palette of green/red/yellow or from the palette cyan/magenta/white. The other colour is the background colour which is selected from the palette of 16 colours.

FUNCTIONAL DESCRIPTION

Figure 5.4 is a block diagram of the M24/M21 Display Controller. The following is a functional description of the major blocks in this controller.

DISPLAY RAM

The display RAM on the Indigenous Display Controller consists of one 16K x 16 word dynamic RAM memory bit plane. It resides in the processor address space, starting at address hex B8000. It is an arbitrated read/write memory, i.e. the processor and the CRT control unit have access to this memory during all modes of operation except when in the 80 by 25 alphanumeric mode. This is done through the bi-directional CPU data buffers and the CRT data buffers.

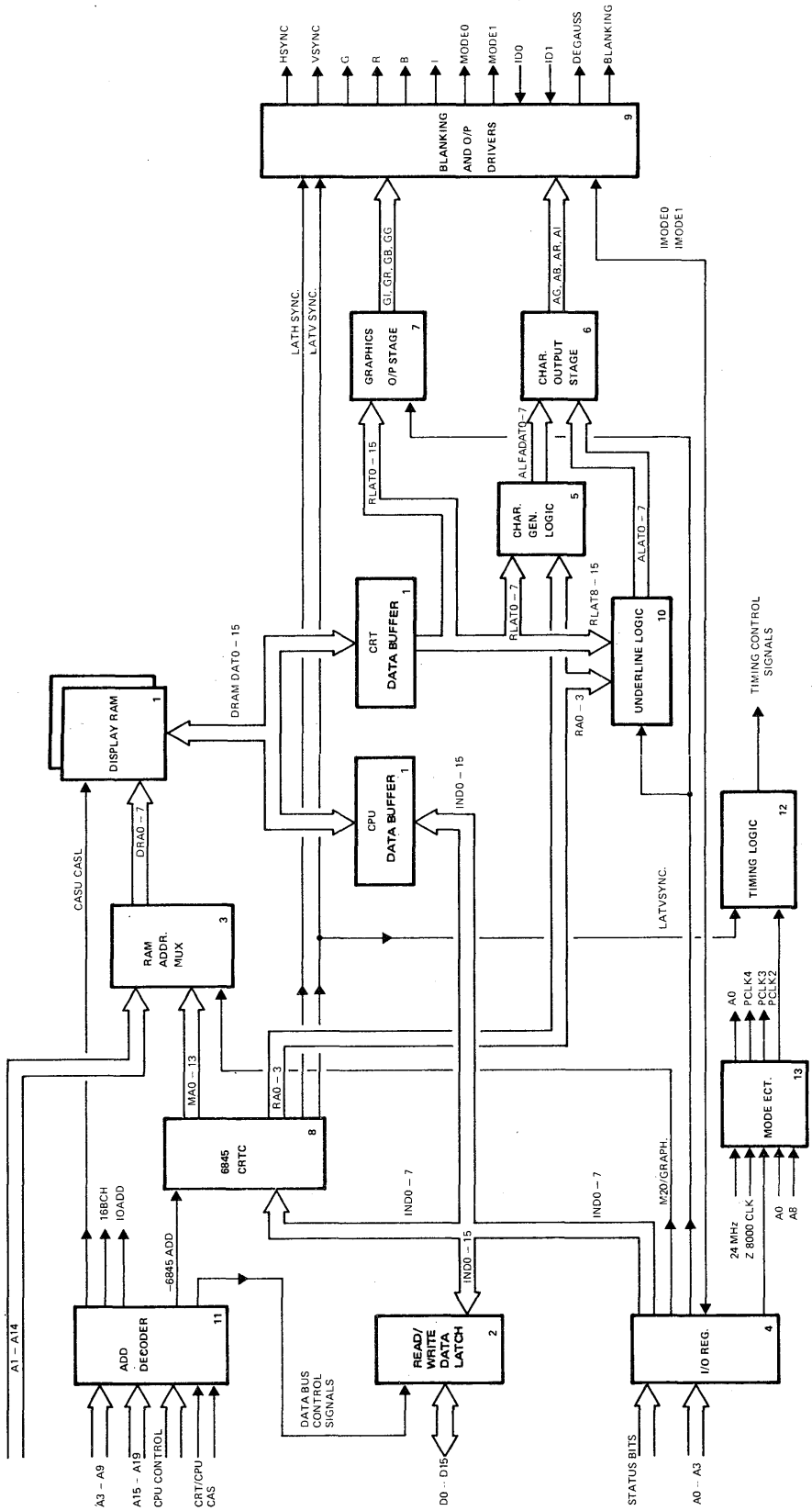


Fig. 5-3 M24/M21 Display Controller Block Diagram

MEMORY ADDRESSING

Memory addressing is done by the DRAM address multiplexer which generates the RAM address signals DRA0-DRA7.

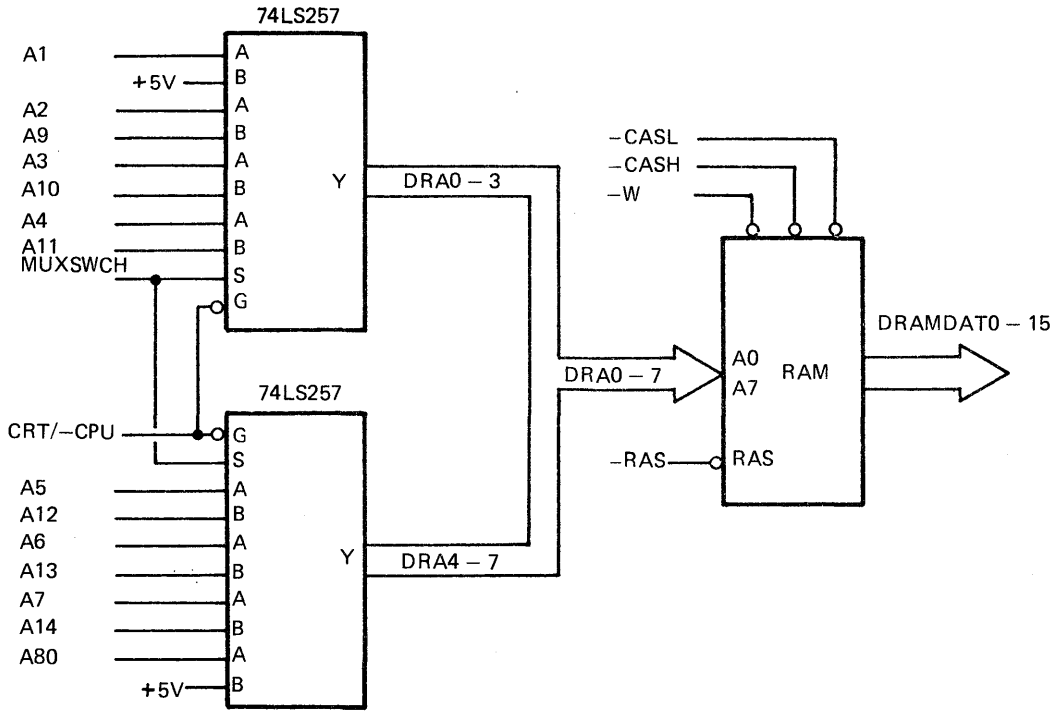


Fig. 5-4 Memory Addressing

The CPU address rows A1-A14 are used to provide the 16K addresses of the 16K x 16 memory bit plane.

The CPU address latch which consists of the two latches 74LS257A is enabled when the signal CRT/-CPU goes low. This is enabled when the CPU wants to read or write into the Display Buffer. The signal MUXSWCH which is generated by the PROM AM27S19A selects which address rows go on the output of the multiplexer. When MUXSWCH is low, A1-A8 go on the RAM address rows DRA0-DRA7.
When the signal -RAS goes low, this address is latched as the row address.

When MUXSWCH is high, A9-A14 go on DRA1-DRA6 rows and when -CASU or -CASL go low, this address is latched as the column address.

The -CASU and -CASL signals determine which byte is to be addressed, the upper or lower byte. These signals are generated by the address decoder circuitry which consists mainly of the PAL 10L8.

DATA BUS BUFFERING

Data from the CPU to the display buffer D0-D15 first passes through the bi-directional Data Bus Buffer, which consists of two 74LS373 buffers and two 74LS244 buffers.

The direction of the data bus buffer is determined by enabling either the LS373 buffer for data towards the CPU or the LS244 buffers for data towards the display buffer.

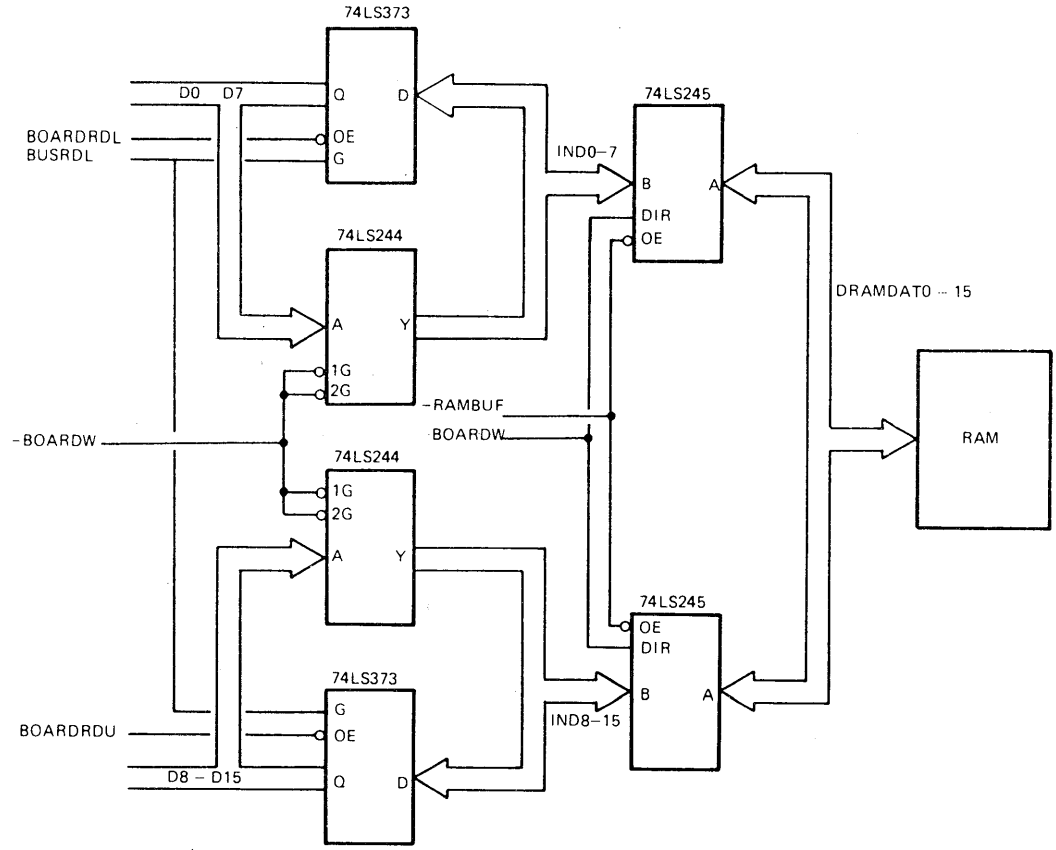


Fig. 5-5 Data Bus Buffering Circuitry

The two LS244 Buffers are enabled by the signal -BOARDW going low. Then the data bits D0-D7 go on to IND0-IND7 and D8-D15 go onto IND8-IND15. The two LS373 buffers are enabled by the -BOARDRDL and -BOARDRDU signals going low. Data is latched by -BUSRDL going high to free memory for use by the CRTC. Thus the data bits IND0-IND7 go on to D0-D7 and IND8-IND15 go on to D8-D15.

Data from the CPU, after passing through the data bus buffer, gets to the CPU data buffer which consists of two LS245 3-state transcievers. These are enabled by the signal -RAMBUF going low which is generated by the address decoder circuitry whenever the CPU is to access the RAM Buffer.

The direction of this data buffer is determined by the signal -BOARDW. When it is low, the direction is towards the RAM buffer and when high, direction is towards the CPU.

CRT DATA BUFFER

The CRT data buffer receives data DRAMDAT0-DRAMDAT15 from the RAM buffer. This latch is enabled continually but DRAMDAT0-15 are latched onto the outputs only on clock transition of the signal CCARNDLY.

DRAM ADDRESS MUX

Besides sorting the CPU address bits to decode the required memory location, the DRAM address multiplexer also deals with the Memory Address MA0-13 signals, and Raster Address RA0-RA4 signals generated by the CRT controller, required to access the display memory.

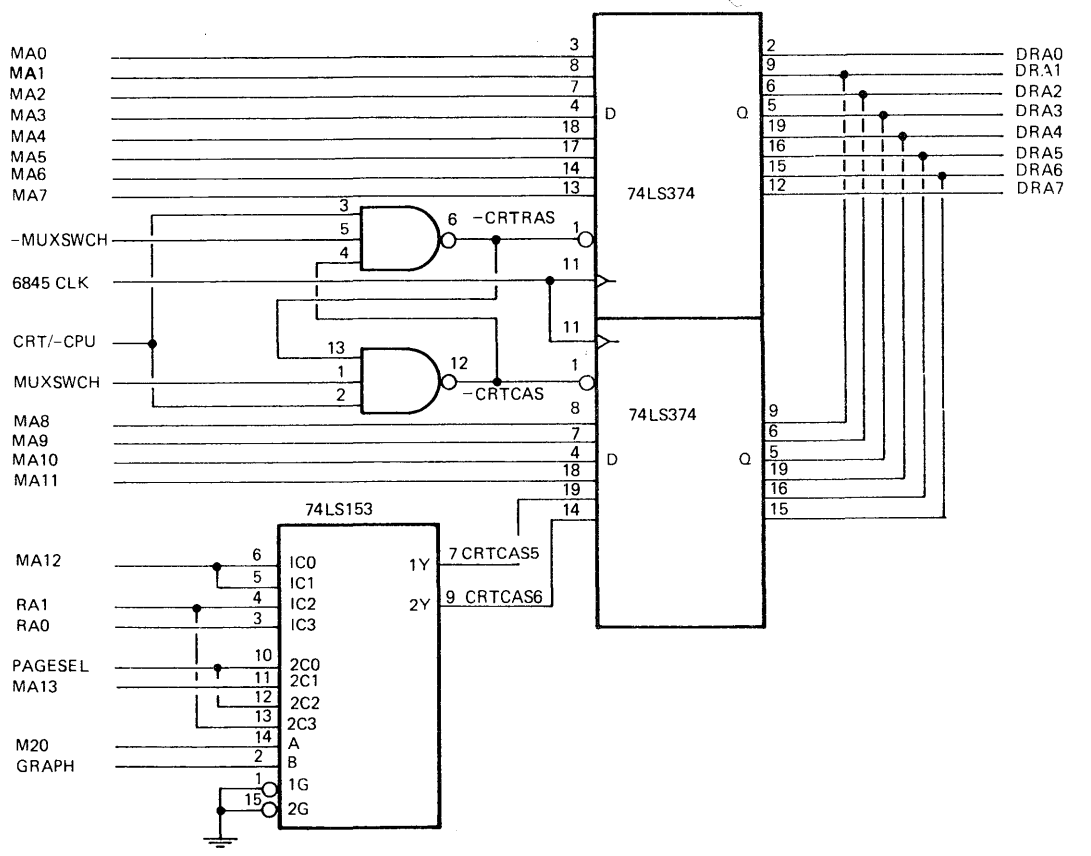


Fig. 5-6 DRAM Address Multiplexer

It consists of two LS374, octal D-type transparent latches, an LS153 dual 4 row to 1 row selector and two NAND gates.
The two NAND gates generate the -CRTRAS and -CRTCAS signals which determine which latch is to be enabled.

When -CRTRAS is enabled, MA0-MA7 are output on the DRA0-DRA7 rows which are then latched into the RAM's as the row address.

When -CRTCAS is enabled, MA8-MA11 plus the two outputs of the selector are output on the RAM address rows as the column address.

The two outputs of the selector depend on the mode of operation as shown in the table below.

MODE	M20	GRAPH	1Y (CRTCAS 5)	2Y (CRTCAS 6)
Alphanumeric	L	L	MA 12	PAGESEL
M20	H	L	MA 12	MA 13
Graphics	L	H	RA 1	PAGESEL
640x400 graphics	H	H	RA 0	RA 1

The signal PAGESEL is generated by the setting of bit 3 of the mode select register 2. When this signal is not active then the 6845 CRT controller have access only to the first 16K bytes of RAM. Setting this bit, access to the remaining 16K bytes of memory is possible.

Setting the M20 bit (bit 0 of mode register 2) when in alphanumeric mode will allow access to all the 16K characters of display memory regardless of the setting of the PAGESEL bit.

I/O REGISTERS

There are various general purpose programmable I/O registers in this display controller. These registers are defined according to the following table.

HEX ADDR.	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0	FUNCTION OF REGISTER
X'3D8'	1	1	1	1	0	1	1	0	0	0	MODE SELECT 1
X'3D9'	1	1	1	1	0	1	1	0	0	1	COLOUR SELECT
X'3DA'	1	1	1	1	0	1	1	0	1	0	STATUS
X'3DB'	1	1	1	1	0	1	1	0	1	1	CLEAR LIGHT PEN
X'3DC'	1	1	1	1	0	1	1	1	0	0	LATCH SET LIGHT PEN
X'3D0'	1	1	1	1	0	1	0	Z	Z	0	6845 REGISTER
X'3D1'	1	1	1	1	0	1	0	Z	Z	1	6845 REGISTER
X'3DE'	1	1	1	1	0	1	1	1	1	0	MODE SELECT 2
X'3DD'											LOOK UP TABLE

Logically this is done on page 4 of the schematics by the decoder LS138. The input signal -IOADD is low whenever the address bits A9-A4 have the value of '3D' hex. This signal, (which is generated on page 11 of the schematics) enables the decoder. Bits A0-A3 are then decoded to select the particular register as shown in the above table.

Mode Select Register 1 (write only)

This is a 6 bit output only register; it cannot be read and its address is '3D8' hex. Its output is as follows:

BIT	NAME	FUNCTION
0	HRES	Logic 1 for 80x25 alphanumeric mode, otherwise 0
1	GRAPH	Logic 1 for all graphics modes, otherwise 0
2	BW	Not connected for M24/M21
3	VIDE	Blanks display when logic 0. 1 enables display signal
4	640x200BW	Logic 1 for 640x400 and 640x200 graphics, otherwise 0
5	BLINK/-BI	Alphanumeric only - 1 for blink, 0 for background I

Bit 5 determines the function of alphanumeric attribute bit 7 and when active, it enables the blink attribute in alpha mode. When low, the blink attribute defines the character background intensity.
The contents of this register are cleared by the system reset signal.

Mode Select Register 2

This is an 8 bit output only register with address '3DE' hex. Its output is as shown in the table below:

BIT	NAME	FUNCTION
0	M20	Active high for Non IBM Modes.
1	DEGAUSS	Active high for degauss request. (Colour monitor degauss)
2	SETSEL	Selects alternate character set when high.
3	PAGESEL	Allows use of 2nd set of screen pages in IBM modes
4	-IMODE0	Inverse of MODE0 signal to monitor
5	IMODE1	MODE1 signal to monitor
6	UNDERLINE	Allows underline in colour alpha modes when high
7	Z8000CLKE	Disable 24MHz clock and enable 19MHz one from Z8000 card

The contents of this register are also cleared by the system reset signal.

Colour Select Register

This is a 6 bit output only register with address '3D9' hex.

BIT	NAME	FUNCTION
0	BLAT	Blue bit. } 320x200 Mode background
1	GLAT	Green bit. } 640x400 Mode foreground
2	RLAT	Red bit. } 640x200 Mode foreground
3	ILAT	Intensifies colour selected in the above modes.
4	ALTBACK	320x200 foreground intensity.
5	320x200 COLORS	Selects 320x200 mode foreground palette.

When bit 5 is set high, the 320x200 mode palette is selected:

Set one Bit 5 = '1'

C1	C0	
0	0	Background Colour (1 to 16)
0	1	CYAN
1	0	MAGENTA
1	1	WHITE

Set two Bit 5 = '0'

C1	C0	
0	0	Background Colour
0	1	GREEN
1	0	RED
1	1	YELLOW

STATUS REGISTER

The status register is an 8 bit read only register with address '3DA' hex.

BIT	NAME	FUNCTION
0	-DISPENDLY1	When high, indicates that retrace is occurring.
1	LPSTB	Lightpen strobe. "1" if lightpen triggered.
2	LPSW	Lightpen switch. "0" is on. Not debounced.
3	LATVSYNC	High during 1st half of vert retrace in alpha mode
4	MONIDO	Monitor ID bit 0
5	MONIDI	Monitor ID bit 1
6	EXPIDO	} Both high no expansion
7	EXPID1	} Both low display option board present

M0N1D0 and M0N1D1 are signal lines coming from the monitor to indicate monitor type as monochromatic or colour according to the following table:

MONIDO	M0NID1	Monitor type
0	1	12" Colour
1	1	12" Monochromatic

6845 Registers

The 6845 address Register is located at addresses 3D0, 3D2, 3D4 and 3D6. This register points to one of 18 programmable registers within the 6845. I/O operations at addresses 3D1, 3D3, 3D5 or 3D7 will read or program the selected register.

ROM CHARACTER GENERATOR

This display controller utilizes a character generator ROM. It consists of 4K or 8K bytes of storage which cannot be read/written under software control.

With 4K bytes of ROM only, one character set is available. With an 8K byte ROM, a second character set is implemented in for various applications.

This ROM is addressed by RA0-RA3 from the 6845 CRTC controller and RLAT0-7 from the display buffer. Its outputs ALFADAT0-7 are used by the character output circuitry.

HD 6845 CRT CONTROLLER

The HD 6845 CRT controller generates the memory addresses, raster addresses, display timing and display enable signals. It consists of 19 accessible internal registers, horizontal and vertical timing circuits, linear address generator, cursor control circuit, and light pen detection circuit.

The 19 accessible internal registers are used to define and control a raster-scan CRT display. The following table defines the values that must be loaded into the 6845 CRT Controller to control the different modes of operation supported by the M24/M21 Display Controller.

The horizontal and vertical timing circuits generate RA0-RA4, DISPTMG, HSYNC, and VSYNC. RA0-RA4 are raster address signals and are used for character generation. DISPTMG, HSYNC, and VSYNC signals are used in the display control circuitry. This horizontal and vertical timing circuit consists of an internal counter and comparator circuit.

In alpha modes, the RA lines are programmed to count repeatedly from 0 to 15 since 16 scan lines per character row are required. They are used to select the correct line of the current character from the character font ROM. They are also used to detect scanline 14 of each character row for creating SETFOREU for character underlining.

In the 200 line graphics modes, the RA lines are programmed to count from 0 to 3. RA0 is then ignored in addressing the display RAM for the display lines to be paired since lines 0 and 1 are identical, as are lines 2 and 3. Thus, effectively, only 200 lines are displayed. For IBM compatibility, RA1 is used as the MSB addressing the DRAM so that the even lines are in the lower part of the RAM while the odd lines are in the upper part.

In the 400 line graphics modes, the RA lines are also programmed to count from 0 to 3 but in this case no pairing is necessary. RA0 and RA1 are used as the MSB's of the DRAM address and so the display RAM is effectively divided into four blocks as described later.

ADDR REG.	REG. #	REGISTER TYPE	UNITS	I/O	40x25 ALPHA	80x25 ALPHA	GRAPHIC MODES
0	R0	Horizontal Total	Char.	Write Only	38	71	38
1	R1	Horizontal Displayed	Char.	Write Only	28	50	28
2	R2	Horiz. Sync Position	Char.	Write Only	2D	5A	2D
3	R3	Horiz. Sync Width	Char.	Write Only	06	0C	06
4	R4	Vertical Total	Char. Row	Write Only	1F	1F	1F
5	R5	Vertical Total Adjust	Scan Line	Write Only	06	06	06
6	R6	Vertical Displayed	Char. Row	Write Only	19	19	64
7	R7	Vert. Sync Position	Char. Row	Write Only	10	10	70
8	R8	Interlace Mode	-	Write Only	02	02	02
9	R9	Max Scan Line Addr.	Scan Line	Write Only	07	07	01
A	R10	Cursor Start	Scan Line	Write Only	06	06	06
B	R11	Cursor End	Scan Line	Write Only	07	07	07
C	R12	Start Addr. (H)	-	Write Only	00	00	00
D	R13	Start Addr. (L)	-	Write	00	00	00
E	R14	Cursor Addr. (H)	-	Read/ Write	XX	XX	XX
F	R15	Cursor Addr. (L)	-	Read/ Write	XX	XX	XX
10	R16	Light Pen (H)	-	Read Only	XX	XX	XX
11	R17	Light Pen (L)	-	Read Only	XX	XX	XX

Fig. 5-7 6845 CRTC Register Description

The linear address generator generates the refresh memory address lines MA0-MA13 which are used for accessing periodically the refresh memory and so refreshing the screen. As 14 refresh address signals are available, 16K words or 32K bytes are accessible. Moreover, the use of the start address register enables paging and scrolling.

The light pen detection circuit detects light pen position on the screen. When a light pen strobe signal is received, the light pen register memorizes the linear address generated by the generator in order to memorize the light pen position on the screen.

The cursor control circuit controls the position of the cursor, its height and its blink.

Fig. 5-7 shows the pin functions of the HD6845 CRT Controller.

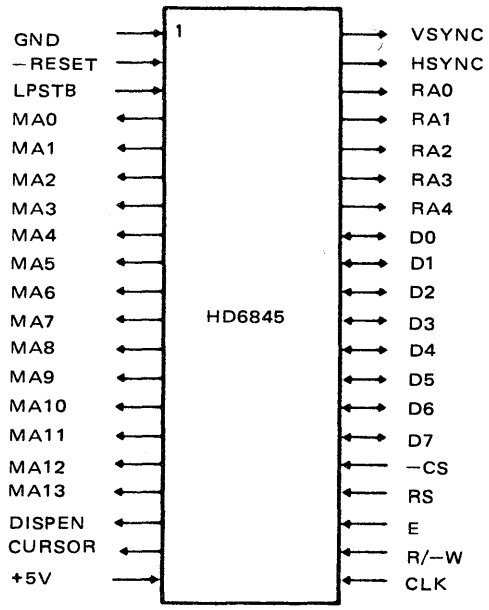


Fig. 5-8 HD6845 Pin Functions

CPU Interface Pins The CRTC interfaces to the CPU bus on the bi-directional bus (D0-D7) using -CS, RS, E and R/-W as control signals.

D0-D7 Data Bus: The 3-state data bus buffer interfaces the HD6845 with the system data bus. Data is transmitted or received on D0-D7 upon execution of an I/O instruction.

E Enable: This signal is used as a strobe signal in a CPU R/W operation with the CRTC internal registers.

-CS Chip Select: A low on this input selects the CRTC to read or write the internal registers. This signal should only be active when there is a stable address being decoded from the CPU.

RS Register Select: This input selects the address register when low and the 18 control registers when high.

R/-W Read/Write: This input selects the direction of the data transfer between the CPU and the CRTC. When high, data is transferred to CPU and when low, data from the CPU is transferred to the CRTC.

CRT Control Signals - The CRTC provides horizontal sync, vertical sync, and Display Enable signals.

VSYNC Vertical Sync: This output is active high and provides the vertical synchronization for the display device.

HSYNC Horizontal Sync: This output is active high and provides the horizontal synchronization for the display device.

DISPEN Display Enable: This output defines the display period in horizontal and vertical raster scanning.

Refresh Memory Addressing Pins

MA0-MA13 Refresh Memory Address: These 14 outputs are used to refresh the CRT screen with pages of data within the 32KB of refresh memory.

RA0-RA4 Raster Address: These 5 outputs are used to select the raster of the character generator.

Other Pins

CLK Clock: The clock input used to synchronize all the CRT control timings.

CURSOR: This output indicates cursor display to the external display processing logic.

-RES: This input is used to reset the CRTC. A low on this input forces the CRTC into the following status:

All the counters in the CRTC are cleared and
the device stops the display operation.

All outputs go to a low level.

Control registers in the CRTC remain unchanged.

The -RES input has the capability of reset only when LPSTB is low. The CRTC starts the display operation immediately after -RES goes high.

SCRAMBLER CIRCUITRY

The 6845 CRT Controller requires some circuitry to translate the data sent to it, by application programs which do not follow the BIOS, into the appropriate data for the M24/M21 display controller. This circuitry, called the Scrambler Circuitry, converts the IBM 6845 register values to the values which the M24/M21 display controller requires.

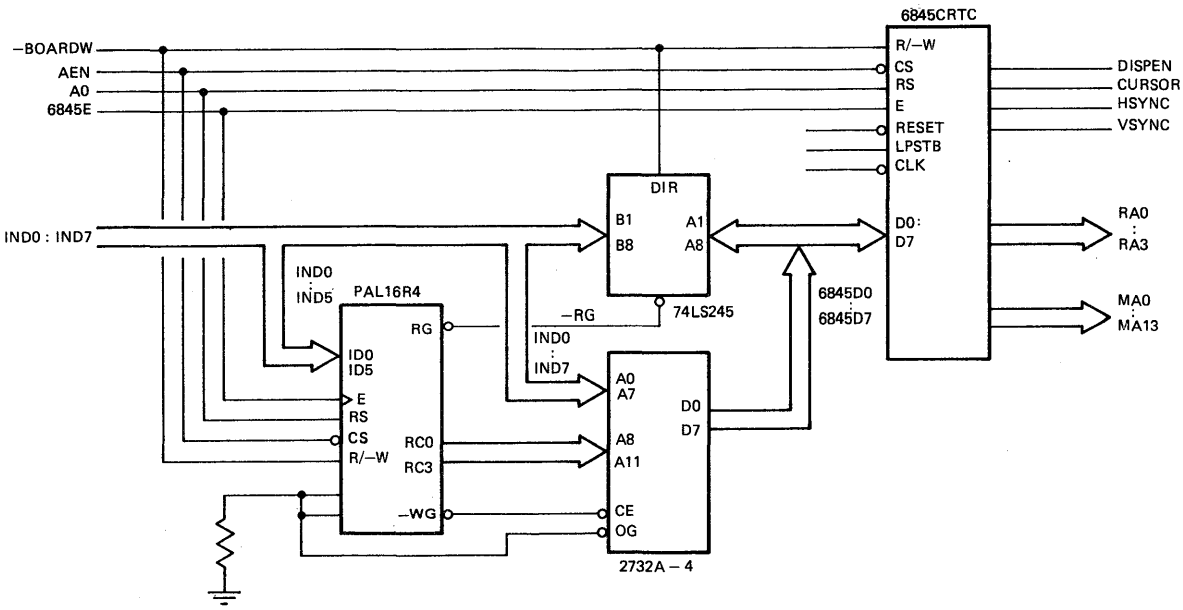


Fig. 5-9 Scrambler Circuitry Block Diagram

The signals that determine whether the data going to the 6845 CRT controller is correct, or whether it needs to be modified, are A0 and -BOARDW. These signals together with the AEN, 6845E and IND0-IND7 signals are input to the PAL 16R4.

The PAL output -RG is low when either -BOARDW is low (memory write operation) or when -BOARDW is high (memory read operation) and A0 is low (not a register select operation). Thus data bits IND0-IND7 travel directly to and from the 6845 controller without any modification.

The PAL output -WG is low (-RG is high), when A0 is high (register select operation) and -BOARDW is high (on a memory read operation). Thus data bits IND0-IND7 are used as address bits A0-A7 to the PROM 2732A. Furthermore, PAL outputs, -RC0 to -RC3, are used as the address bits A8-A11 for the PROM. In this case the data used by the 6845 controller is the contents of the addressed location in PROM. The contents of this PROM are found in appendix A.

The PAL 16R4 outputs are according to the following equations.

-RG = E*-CS*W*-TST1 + -W*-RS*-CS*-TST1

-WG = -CS*-W*RS*-TST1

-RC0 = -RC0*RS + -RC0*-CS + -ID0*-RS*-CS*-ID4*-ID5

-RC1 = -RC1*RS + -RC1*-CS + -ID1*-RS*-CS*-ID4*-ID5

-RC2 = -RC2*RS + -RC2*-CS + -ID2*-RS*-CS*-ID4*-ID5

-RC3 = -RC3*RS + -RC3*-CS + -ID3*-RS*-CS*-ID4*-ID5

ALPHANUMERIC MODES

Every display character position is defined by two bytes in the display controller memory, namely the attribute byte and the character code byte.

ATTRIBUTE BYTE								CHARACTER CODE BYTE							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ODD ADDRESS								EVEN ADDRESS							

When in the alphanumeric mode, the display controller have the following characteristics:

Display up to 25 rows of either 40 or 80 characters each
Character cell size of 8 x 16 pixels without display option board
Character cell size of 16 x 16 pixels with display option board
Requires 4000 bytes of controller RAM for 80 column mode
Requires 2000 bytes of controller RAM for 40 column mode
Eight pages may be stored for 80 column text
Sixteen pages may be stored for 40 column text

The starting address of the display buffer is at the location B8000 hex and is 32K bytes long.

The HRES bit of mode select register 1 defines the alphanumeric mode used. The 40x25 mode is selected when HRES=0 and the 80x25 mode when HRES=1

If BLINK/-BI and UNDERLINE bits are both 0, then the attribute byte is defined as follows:

15	14	13	12	11	10	9	8
I	R	G	B	I	R	G	B
BACKGROUND				FOREGROUND			

Thus one of sixteen colours (or shades of grey) may be chosen for foreground and background for each cell.

If BLINK/-BI bit is set, the background intensity bit is forced off. Then blinking is controlled by bit 15 of the attribute byte.

The display controller also allows characters to be underlined by setting the UNDERLINE bit. If the attribute byte is set for a blue foreground colour then the resulting character is white and underlined if the UNDERLINE bit is set and blue or dark grey if it is not.

CHARACTER OUTPUT STAGE

RLAT0-RLAT7 pass through the character generator ROM to generate ALFADAT0-7. This is parallel loaded into the shift register S299 when the signal SHFLD goes high and the register clocked. The data bits are shifted out of the QH' output with every clock, bit 7 first, when the SHFLD signal goes low. If the UNDERLINE signal is not active, then RLAT8-14 go directly to the latch S374. RLAT15 is NANDed with BLINK/-BI to determine if background intensity is on or not.

If UNDERLINE is on, then RLAT8-RLAT10 go to the inputs of LS138 where output -UNDCHAR goes low if RLAT8 is low and thus AALAT0-AALAT2 go high. These bits, together with RLAT11-14 and AALAT7, go to the inputs of the latch LS374. The outputs of this latch are ALAT0-ALAT7. ALAT0-ALAT3 constitute the fully decoded foreground colour while ALAT4-ALAT8 the fully decode background colour.

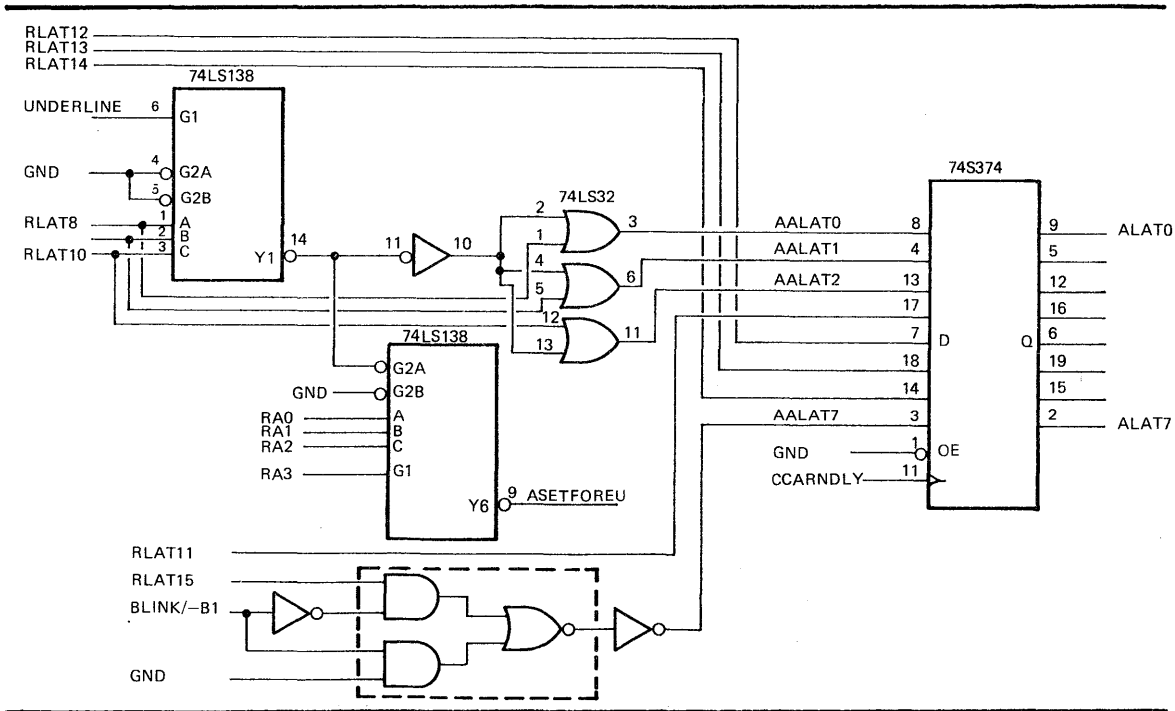


Fig. 5-10 Underline Logic

With UNDERLINE on, RA0 low and RA1-RA3 high, the output ASETFOREU of LS138 goes low and thus the signal SETFOREU goes low. This will select the background attribute bits ALAT4-ALAT7 to generate AB, AG, AR, AI signals for the background colour. The signal FORE/-BACK determines whether the colour information for each pixel is for the background or foreground character. This signal is high for foreground and low for background.

GRAPHICS MODES

The following is a description of the various graphics modes supported by this display controller.

320 x 200 Graphics Mode

In this mode each pixel colour may be one of the four preselected colours. The background colour (colour 0) is defined by the setting of the colour select register bits 0-3. Three other colours may be chosen from two colour sets using bit 5 of the colour select register. Each pixel is defined by two bits and so the data format is as follows:

7 6	5 4	3 2	1 0
C1 C0	C1 C0	C1 C0	C1 C0
1st Display Pixel	2nd Display Pixel	3rd Display Pixel	4th Display Pixel

C1 and C0 selects 4 of 16 preselected colours according to the following table:

C1	C0	Palette Colour
0	0	Background colour
0	1	Colour 1
1	0	Colour 2
1	1	Colour 3

The two colour sets are:

Set one
Colour 1 CYAN Colour 2 MAGENTA Colour 3 WHITE

Set two
Colour 1 GREEN Colour 2 RED Colour 3 YELLOW

Note that the ALTBACK bit can highlight all foreground colours when bit is on.

Graphics Storage Map

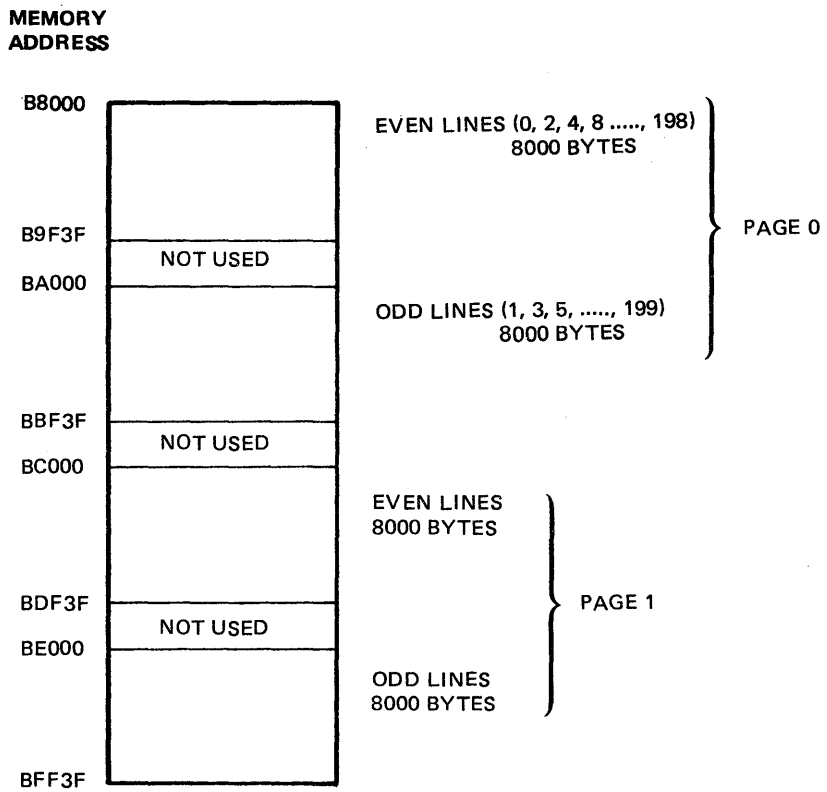


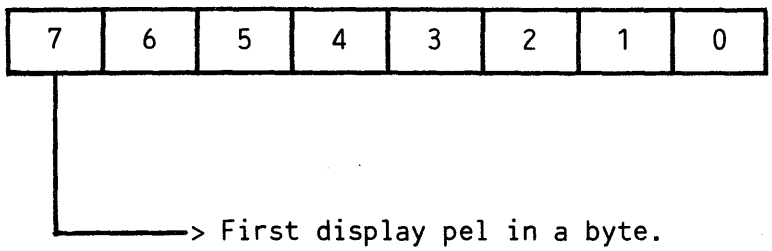
Fig. 5-11 Memory Map (320 x 200 Graphics Mode)

The above table shows the memory map for the 320 x 200 graphics mode. Address B8000 contains pixel information for the upper left corner of the display and are as described previously.

This mode is IBM compatible, but with two pages available instead of one. For the second screen page to be accessed the PAGESEL bit in mode register 2 must be set.

640 x 200 Graphics Mode

This mode is also fully IBM compatible and the addressing and mapping is the same as for the 320 x 200 mode described above.



In this mode only one bit in memory is used for every pixel on the screen and so each memory byte represents eight pixels. Bit 7 of each byte defines the first display pixel while bit 0 defines the last pixel to be displayed for that byte. The background colour is always black. The foreground colour may be one of sixteen colours or shades of grey as chosen by bits 0-3 of the colour select register.

640 x 400 Graphics Mode

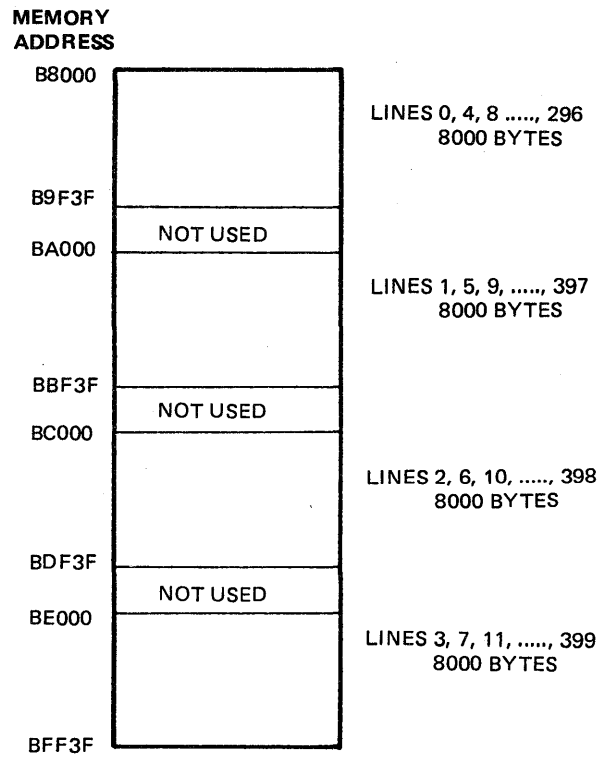


Fig. 5-12 Memory Map (640 x 400 Graphics Mode)

The above table shows the memory map for the 640 x 400 mode. The display buffer begins at address B8000 Hex and the memory size is 32K bytes. This allows for only one page of 640 x 400 graphics. The memory is organized as 4 segments of 8K each. Each segment contains every fourth line beginning with line 0 at B8000 Hex, line 1 at BA000 Hex, line 2 at BC000 Hex, line 3 at BE000 Hex.

The data format is as in the 640 x 200 mode with 1 bit per pixel. In this mode the background colour is black with a software selectable single colour or grey level foreground.

512 x 256 Graphics Mode

This mode is only available when the Z8000 board is connected on to the bus converter board. This mode allows full PCOS (M20 operating system) compatibility. Switching from PCOS to M24/M21 operating system is handled by software.

This mode is the same as the 640 x 400 mode with the exception of resolution and the memory map. The memory map consists of two pages with the first page beginning at address B8000 hex and the 2nd page beginning at address BC000 hex.

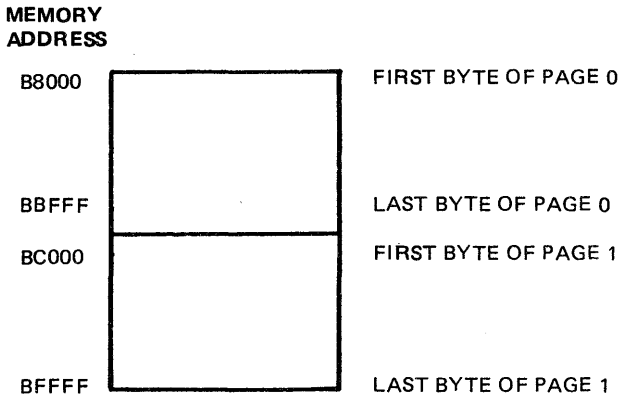


Fig. 5-13 Memory Map (512 x 256 Graphics Mode)

GRAPHICS OUTPUT STAGE

The attributes signals RLAT0-RLAT15 from the display buffer are parallel loaded into the two shift registers 74S299 when the signal GRAPHSHFLD is high and when a clock transition from low to high of the signal PCLK4 occurs. This loaded data is then shifted serially to the outputs 'QH' of the two shift registers. Bits RLAT7 and RLAT6 go out first. Thus the signals ODDGRAPH and EVENGRAPH represent the C1 and C0 bits described earlier and are used to select the pixel colour.

These two bits pass through the latch 74S374 and together with the two signals ALTBACK and 320x200Colors from the I/O registers form the B inputs to the 2 to 1 quad multiplexer 74LS157.

The A inputs to the multiplexer are the colour select bits from the I/O registers.

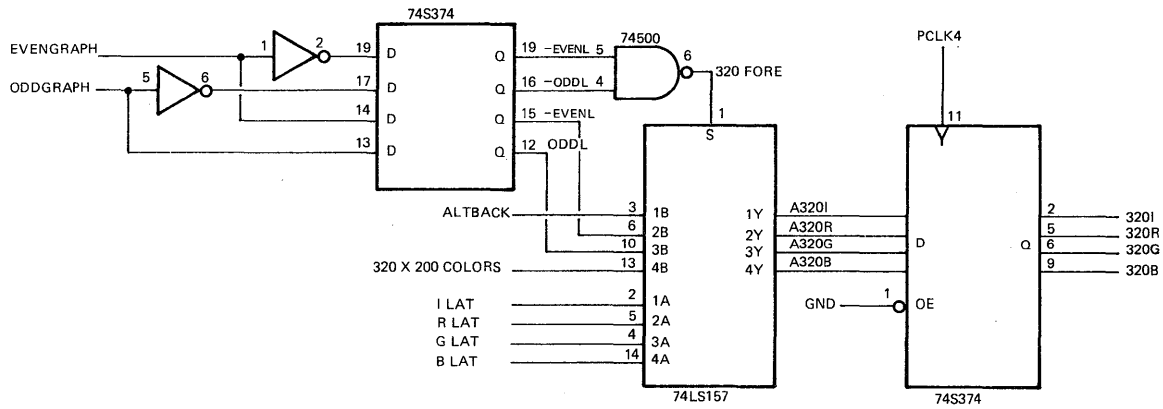


Fig. 5-14 Low Resolution Graphics O/P stage

In the 320x200 mode this multiplexer selects the foreground and background colours by selecting either A or B inputs. The selection is done by the signal 320FORE which is generated by the two signals ODDGRAPH and EVENGRAPH after passing through the 'D' latch and the NAND gate 74S00. When both are low, the signal 320FORE is low and the A inputs are output on to the 4 outputs. These outputs thus constitute the pixel background colour.

When the 320FORE signal is high the B inputs are selected on to the 4 multiplexer outputs and then constitute the pixel colour.

After passing through the 'D' latch 74S374, these 4 outputs go on to the 2A inputs of the output line drivers.

In the 640x200 BW high resolution mode the colour select bits from the I/O registers are loaded into the inputs of the 4 bit synchronous counter 74S163.

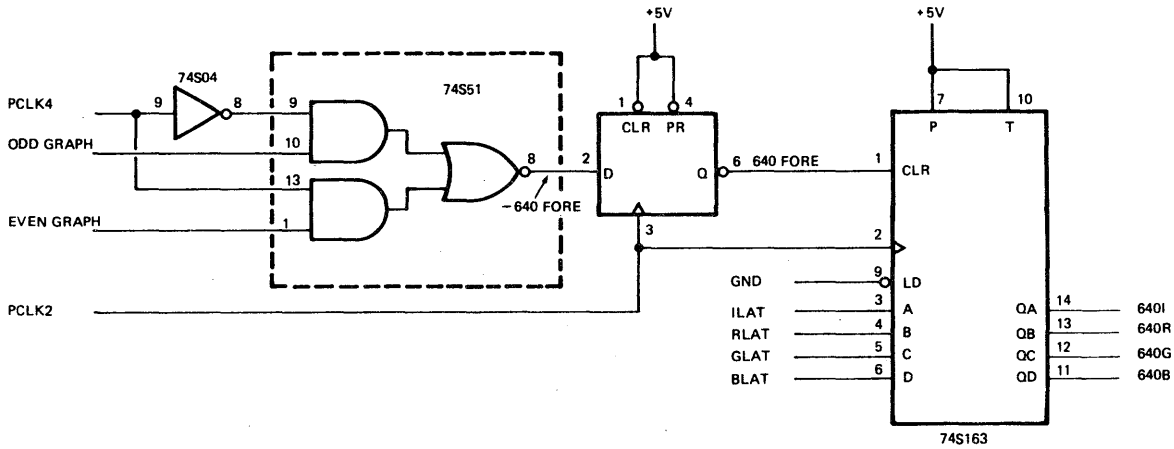


Fig. 5-15 High Resolution Graphics Output Stage

This counter is controlled by the clear input signal 640FORE. The gates 74S51 act as a multiplexer with PCLK4, which is twice as fast as PCLK2, selecting EVENGRAPH or ODDGRAPH. A high on one of these signals will set 640FORE high and the outputs of the counter are determined by the inputs from the colour register. These constitute the foreground pixel colour. A low on both attribute signals set the 640FORE signal low which clears the counter and forces the outputs all low. This gives a black background colour to the pixel.

The 4 outputs of this counter go to the 1A inputs of the output line drivers.

The line drivers 74S244 select the 640 inputs or the 320 inputs according to the enable input signals. If -320x200 is low then the 320I, R, G, B signals are output and if the -640x200BW is low, the 640I, R, G, B are output to give the GI, GR, GB and GG signals which constitute the display signal to be output to the CRT display.

BLANKING AND OUTPUT DRIVER

The GI, GR, GG, GB outputs from the graphics output stage or the AI, AR, AG, AB outputs from the character output stage are loaded into the counter 74S163.

The graphics or alphanumeric display signals are output when clocked by the PCLK3 signal. These outputs go to the enhancement connector and to the latch 74S374 whose outputs constitute the four display signals, namely: Highlight, R, G, B.

When the blanking signal -BLANKING is low the outputs of the counter are forced to zero and thus all the display signals are low. This signal thus creates the blanking effect and the pixel to be displayed is black.

The following table shows the colours possible with the I,R,G,B signals. Note that "I" is the Highlight signal and provides the extra brightness to each shade available.

I	R	G	B	COLOUR	SHADE OF GREY	
0	0	0	0	Black	Black	0
0	0	0	1	Blue	Darkest Grey	1
0	0	1	0	Green		2
0	0	1	1	Cyan		3
0	1	0	0	Red		4
0	1	0	1	Magenta		5
0	1	1	0	Brown		6
0	1	1	1	Light Grey	Lightest Grey	7
1	0	0	0	Dark Grey		8
1	0	0	1	Light Blue		9
1	0	1	0	Light Green		10
1	0	1	1	Light Cyan	Highlight of	11
1	1	0	0	Light Red	0 - 7	12
1	1	0	1	Light Magenta		13
1	1	1	0	Yellow		14
1	1	1	1	White		15

ADDRESS DECODER

The address decoder circuitry decodes the address bits A3-A19 from the CPU together with some CPU control signals to generate the memory control signals, the I/O registers enable signal and the 6845 controller enable signal.

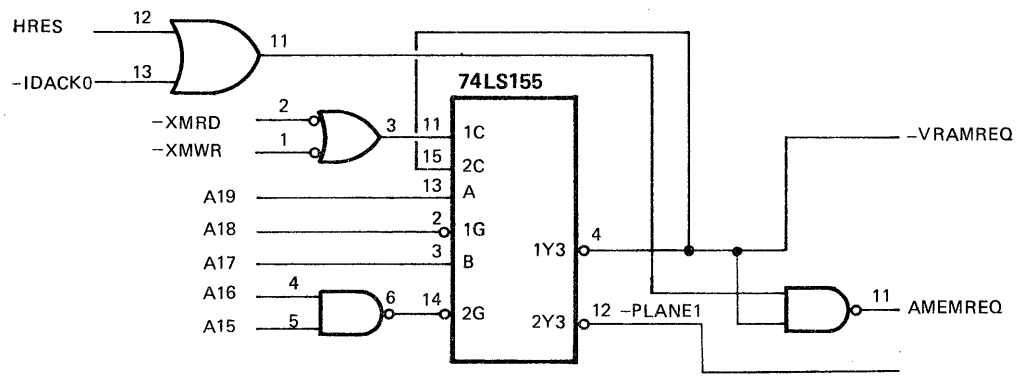


Fig. 5-16 Address Decoder Circuitry

-VRAMREQ is active when the CPU is signalling a memory read/write (-XMRD or -XMWR active) and the address bits A19, A17 are high while A18 is low.

-PLANE1 is active when -VRAMREQ is low and the address bit A19, A17, A16, A15 are all high, while A18 is low.

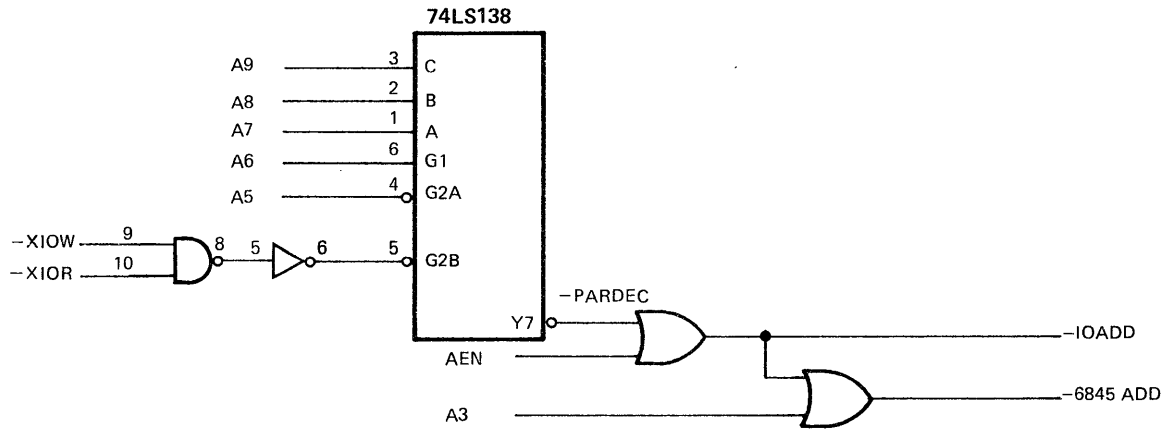


Fig. 5-17 Address Decoder Circuitry

For -IOADD to be active one or more of the signals -XIOW, -XIOR are low and the address bits A5-A9 and AEN signal are as in the table below:

A9	A8	A7	A6	A5	AEN
1	1	1	1	0	0

-6845ADD is active when -IOADD is active and the address bit A3 is low.

-16BCH is active when either -PLANE1 or -IOADD is low.

This address decoder also consists of the PAL 10L8 which generates the memory control signals. The contents of this PAL are found in appendix A.

TIMING LOGIC

The timing logic consists mainly of the PROM AM27S19A which generates the clocks and strobe signals required in this display controller. The PROM is addressed by the outputs of the counter 74F191 and the I/O mode register bit HRES. The counter is clocked by the clock PCLK2.

Fig 5-16 shows the timing diagram of the various signals generated by the timing logic. The PROM contents are found in appendix A.

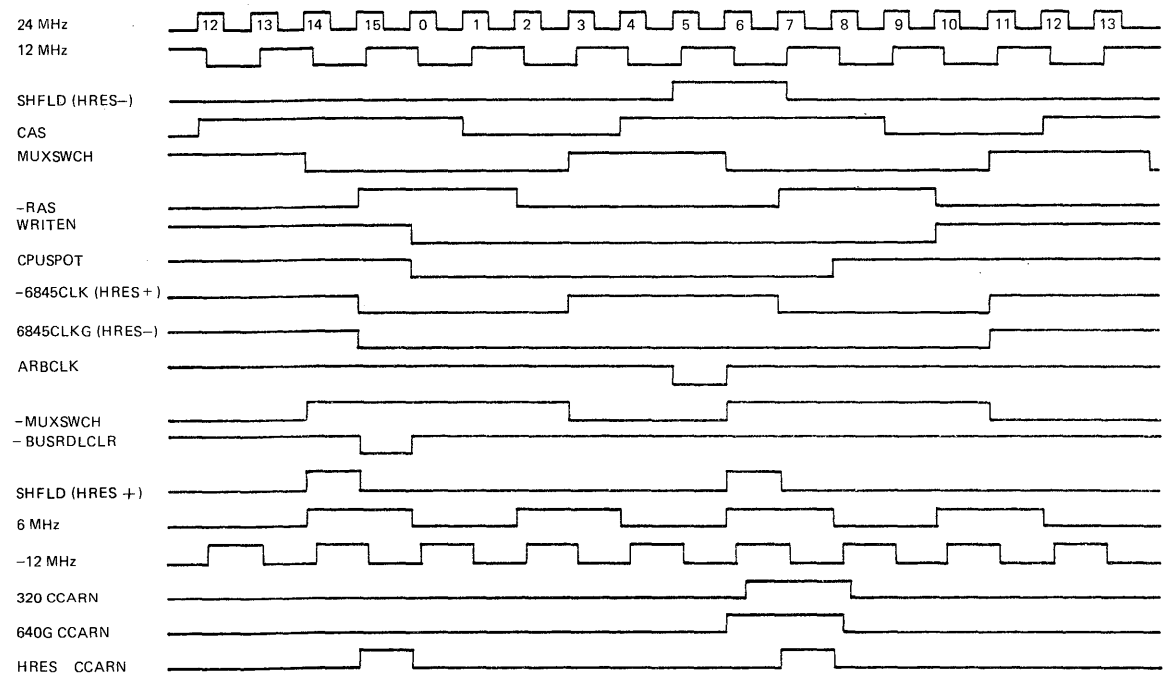


Fig. 5-18 Timing Diagram

The following is a description of the signals shown in the above timing diagram.

SHFLD/ -SHFLD — These initial load shift signals are used as load pulses in various parts of the circuitry.

6845CLK/ -6845CLK — These signals are used as the character rate clocks for the 6845 CRTC. 6845CLK is also used as the latch clock for the CRTC DRAM address latches.

MUXSWCH — The DRAM address multiplexer switch is used to multiplex between row and column addresses for the DRAM.

WRITEN — This signal provides the proper timing for a DRAM write.

CAS — This initial DRAM column address strobe is used to produce the -CASL and the -CASU signals for the DRAM column address.

CPUSPOT This is the CPU enable timing window signal and is high during the time that the CPU is allowed to access the DRAM. It is also used to create ARBCLK and to enable CRT/-CPU low.

ARBCLK This arbitration clock consists of bursts of 12MHz during the low time of CPUSPOT. It is used to latch CPU DRAM access signal MEMWAIT.

RAS This DRAM row address strobe is used directly as the RAS signal for the display DRAM.

CCARN This character and graphics latch / load signal is a positive pulse generated from the main timing signal -SHFLD and the character pixel clock PCLK3.
In character modes, CCARN latches the attribute signals BLINKBIT and SETFOREU so as to synchronize these bits with the character bitstream RAWCHAR.
In 640 wide graphics, CCARN is used as the graphics shift load GRAPHSHFLD to delay the graphics bitstream by one 24MHz cycle which is required for it to be synchronized with the display blanking signal.

MODE CIRCUIT

The mode circuit generates the various clocks used in the display controller by the various modes of operation.

It uses a 24MHz clock as the basic clock, and the clocks generated are:

PCLK2: A 24 MHz clock used mostly in the graphics output stage.

PCLK3: This clock is either the system clock (24MHz in normal modes and 19MHz in the 512 x 256 mode) or a -12MHz clock used mainly in the character output stage, the blanking circuit and the underline logic. The selection of one clock or the other depends on the mode bits HRES or 640x200 BW.
If both bits are low then PCLK3 is the -12MHz clock but if one or both are high then PCLK3 is the system clock.

PCLK4: A 12MHz or a -12MHz clock used in the graphics O/P stage.

GRAPHSHFLD: This is either the clock SHFLD generated by the timing logic or the signal CCARN. This clock is used in the the graphics output stage.

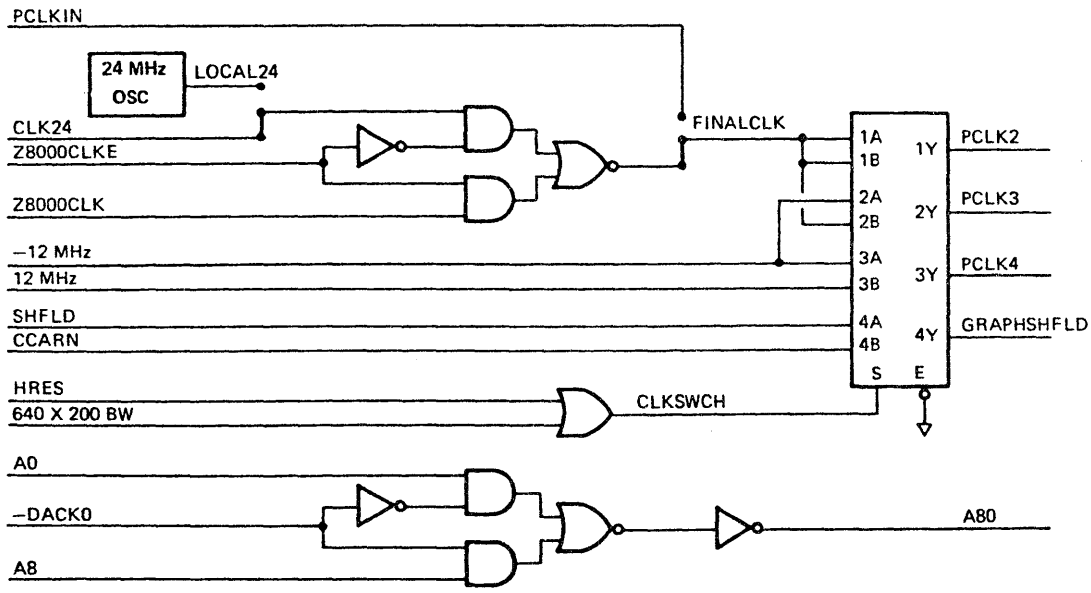


Fig. 5-19 Mode Circuit

The mode circuit also decodes the two address bits A0, A8 and the control bits -IDACK0, Z8000CLK and Z8000CLKE to generate the signals A80 and MUXTCLK.

A80 follows A0 when -IDACK0 is low, and A8 when -IDACK0 is high.

FINALCLK is the main clock to be used by the controller. The selection of the source for this clock is determined by the settings of two jumpers and the mode select register 2 bit Z8000CLKE.

Normally, the jumpers are set in positions 2-3 and 5-6 and the Z8000CLKE bit is set to zero. This causes the signal FINALCLK to be MUXTCLK which in this case is the system 24 MHz clock.

If the APB Z8000 board is present and the Z8000CLKE bit is set high, then the FINALCLK and MUXTCLK are a 19MHz clock.

If the jumper is set to position 1-2, then the source of the 24 MHz clock for the FINALCLK is the 24 MHz oscillator present on the controller. If the jumper is set to position 4-5 the FINALCLK consists of the pixel clock input PCLKIN.

Jumper A	Z8000CLKE	Function
5 to 6 2 to 3	0	Selects system 24 MHz clock, CLK24.
5 to 6 1 to 2	0	Selects crystal oscillator output, LOCAL24.
5 to 6	1	Selects Z8000 clock, Z8000CLK.
4 to 5	x	Selects pixel clock input, PCLKIN.

DEGAUSSING

When DEGAUSS (bit 1 of the mode select register 2) goes active high the screen is degaussed. This is used to cancel the effect of a change in the magnetic field especially after physical movement. This degaussing is only valid for colour monitors and for it to be done properly the display should first be disabled, the degauss bit toggled to produce a pulse of between 1 and 10 microseconds and the display enabled about 5 seconds later.

This degaussing feature also occurs automatically during system start up.

TIMING DIAGRAMS

The following are timing diagrams for the various modes of operation of the display controller board. These timing diagrams refer to rev. P2 boards. Only slight changes occur with rev. P3 boards.

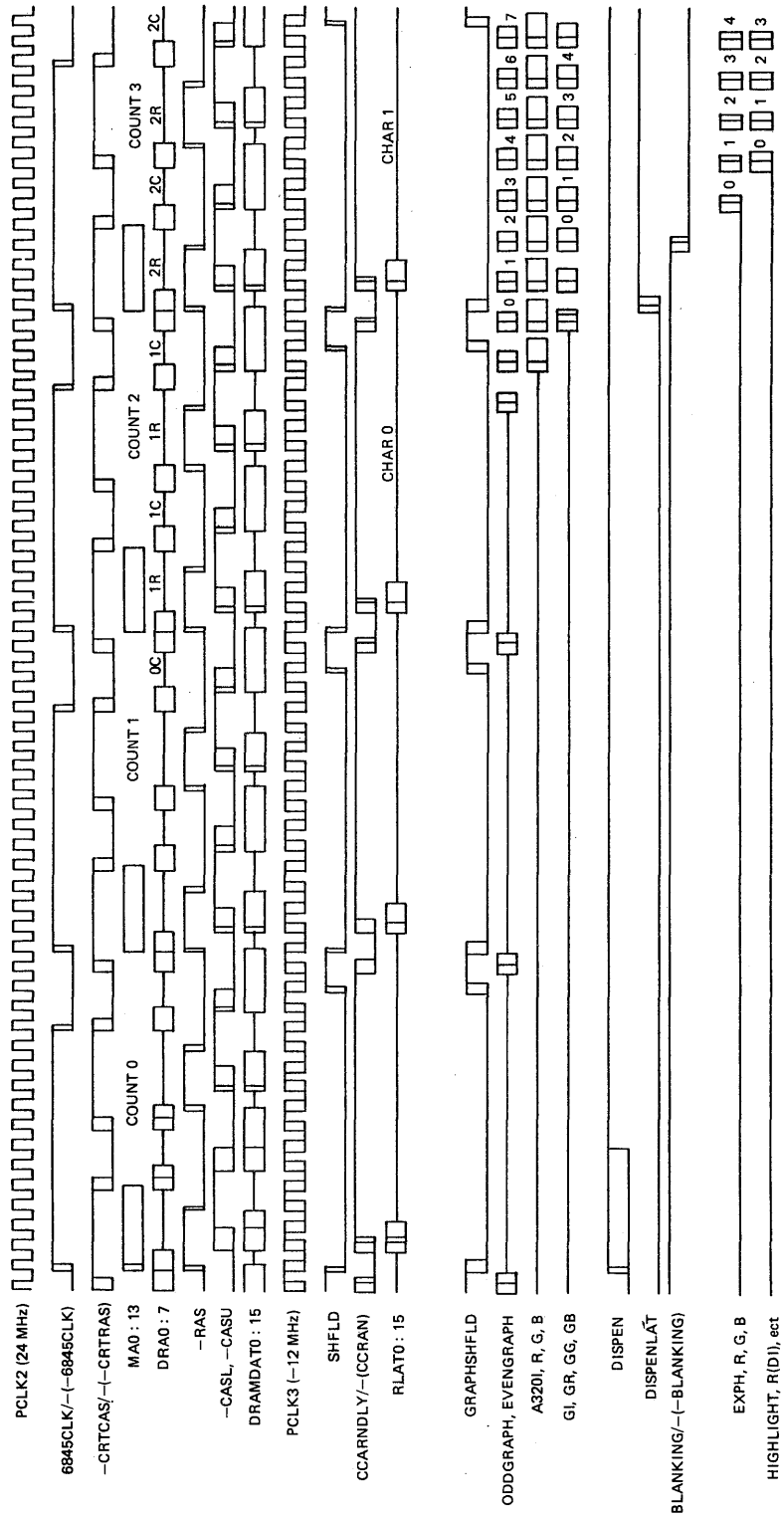


Fig. 5-20 320 Wide Graphics Modes Timing Diagram

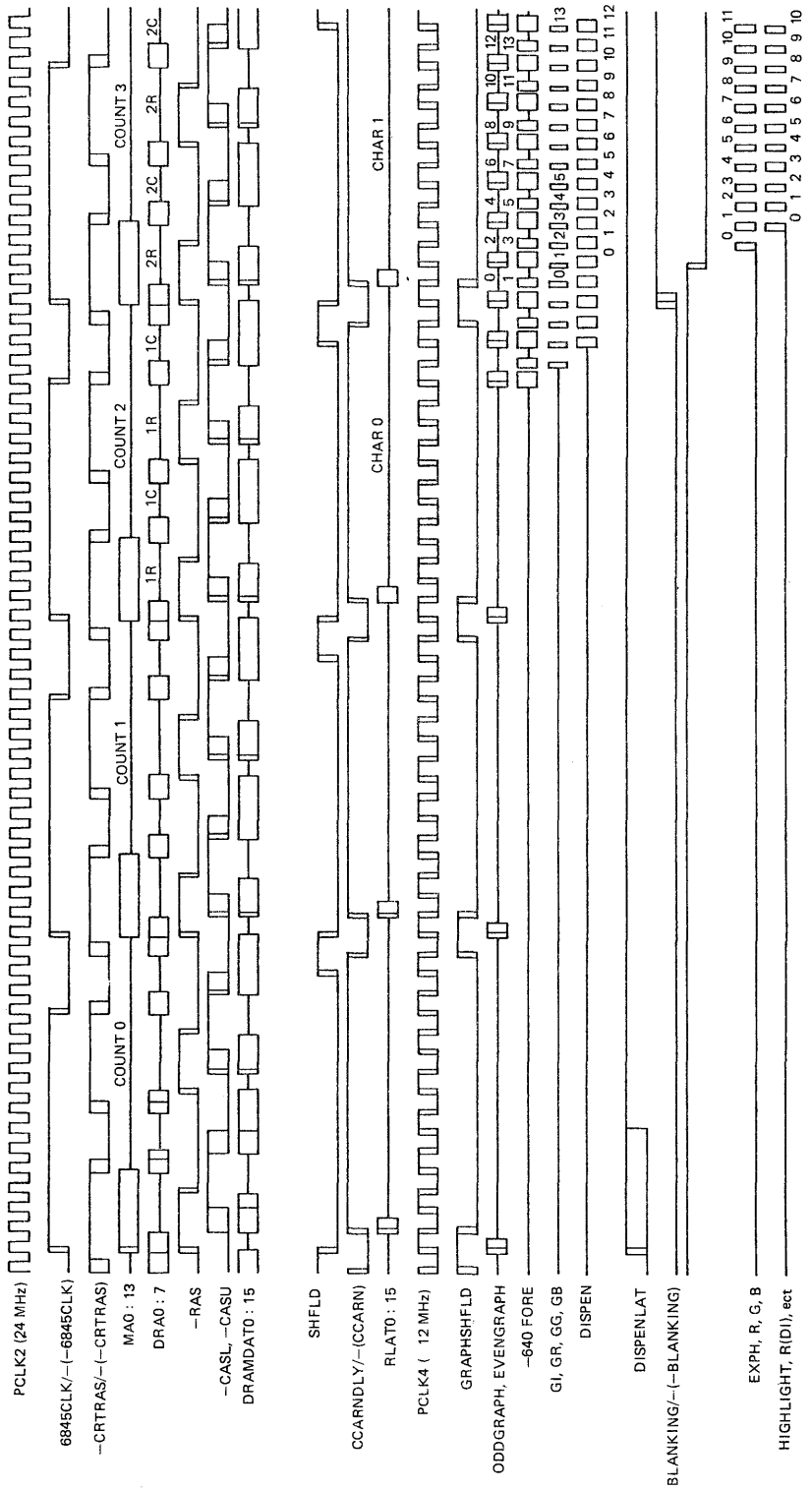


Fig. 5-21 640 Wide Graphics Modes Timing Diagram

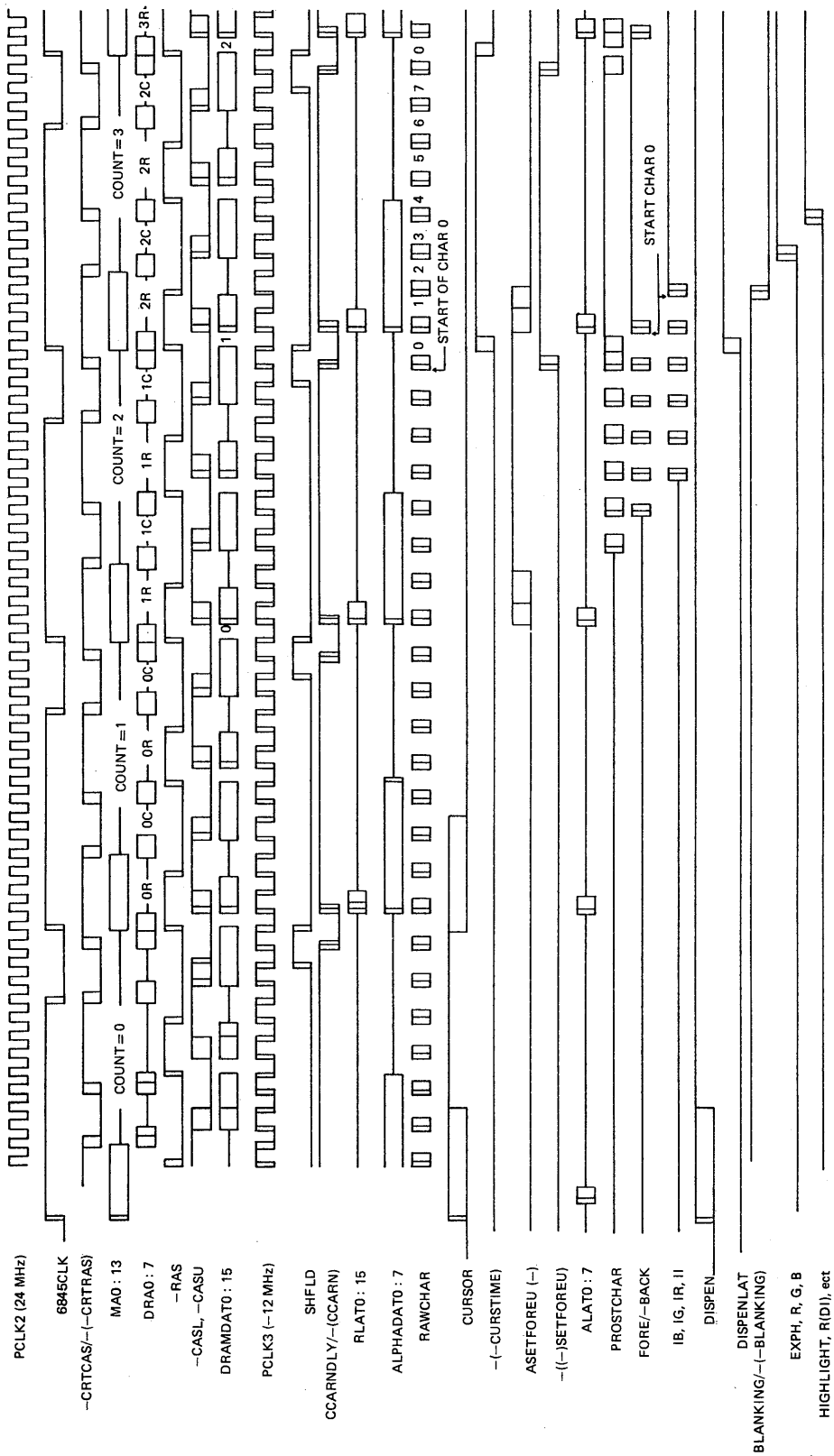


Fig. 5-22 40x25 Alphanumeric Mode Timing Diagram

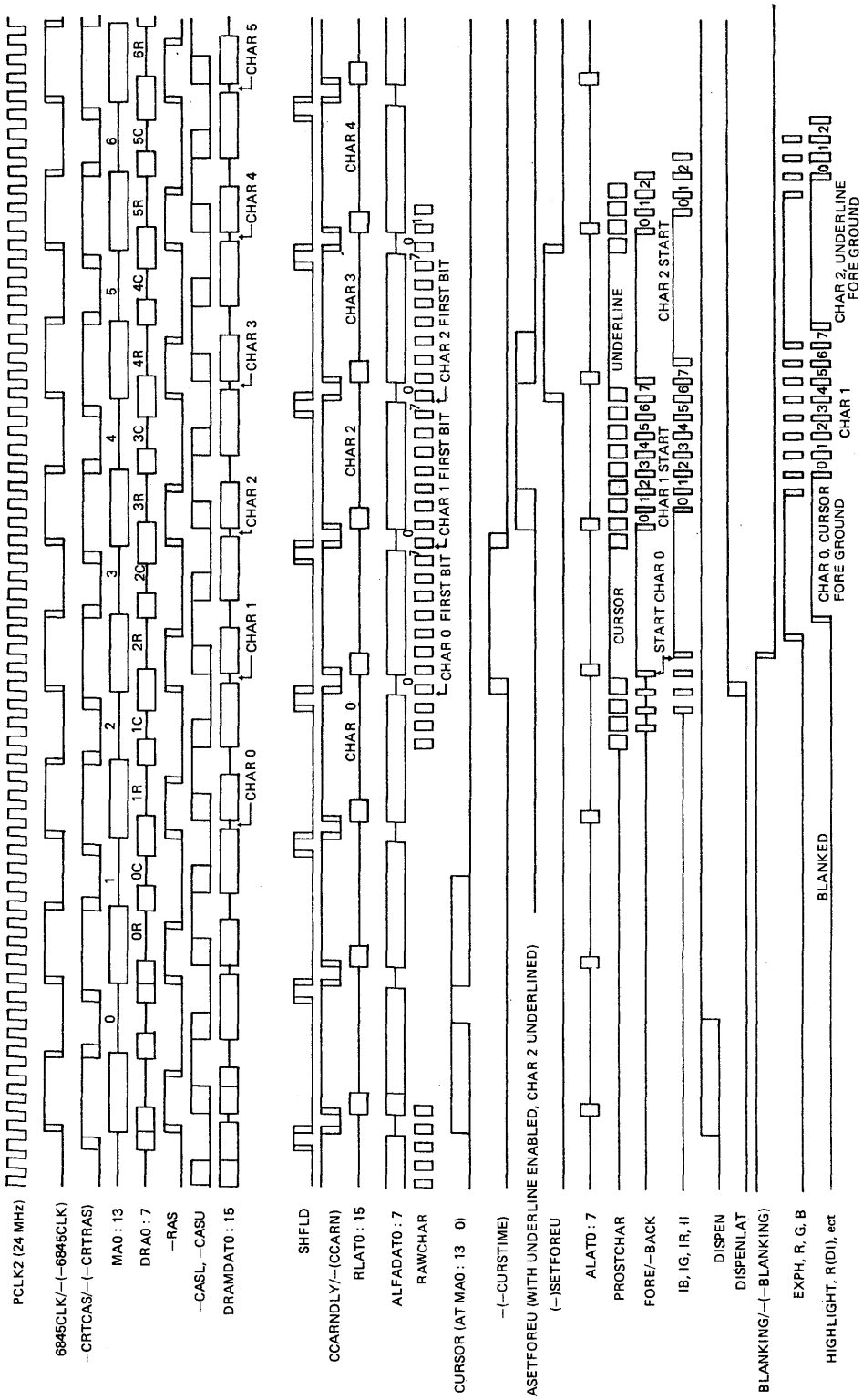


Fig. 5-23 80x25 Alphanumeric Mode Timing Diagram

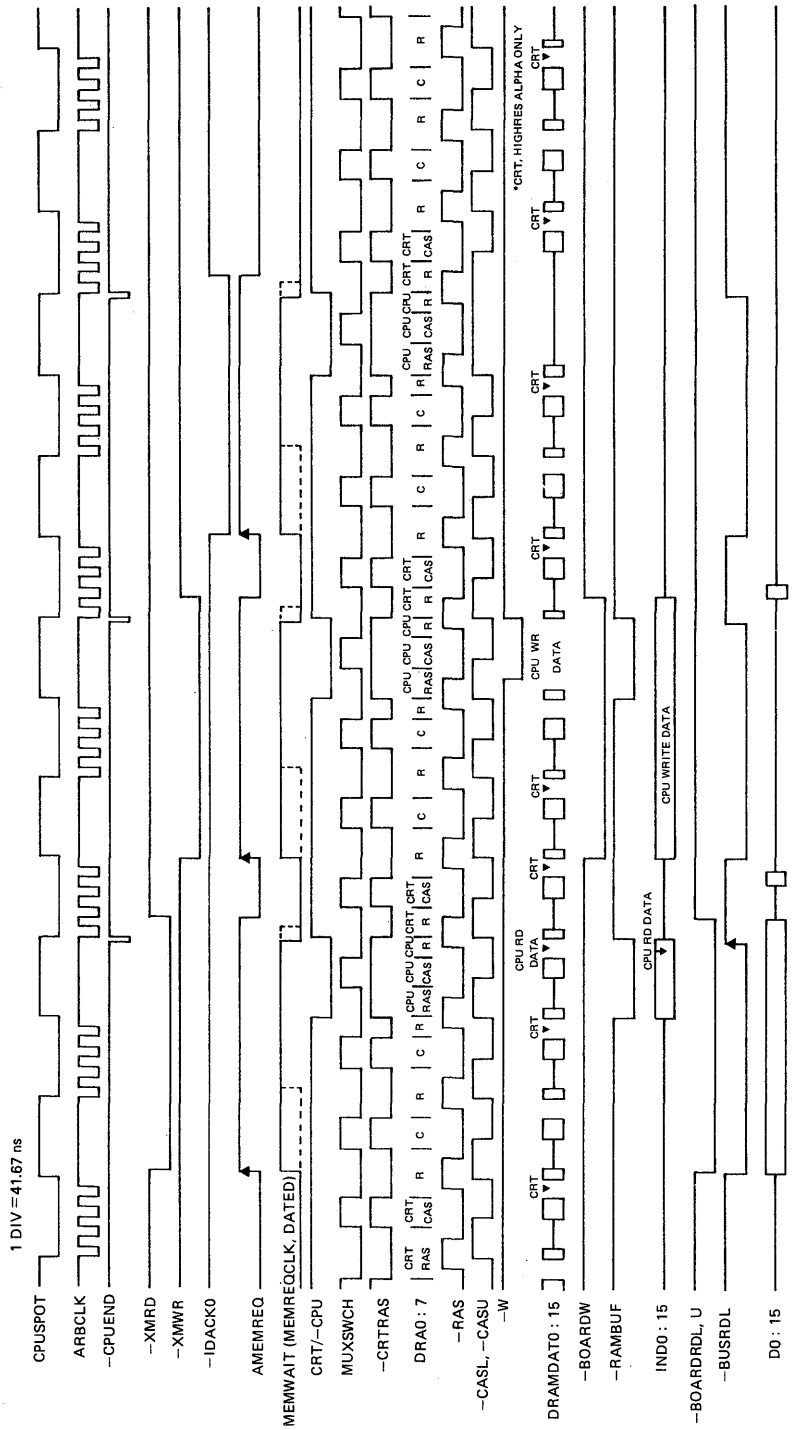


Fig. 5-24 Memory Read/Write/DMA Refresh Timing Diagram

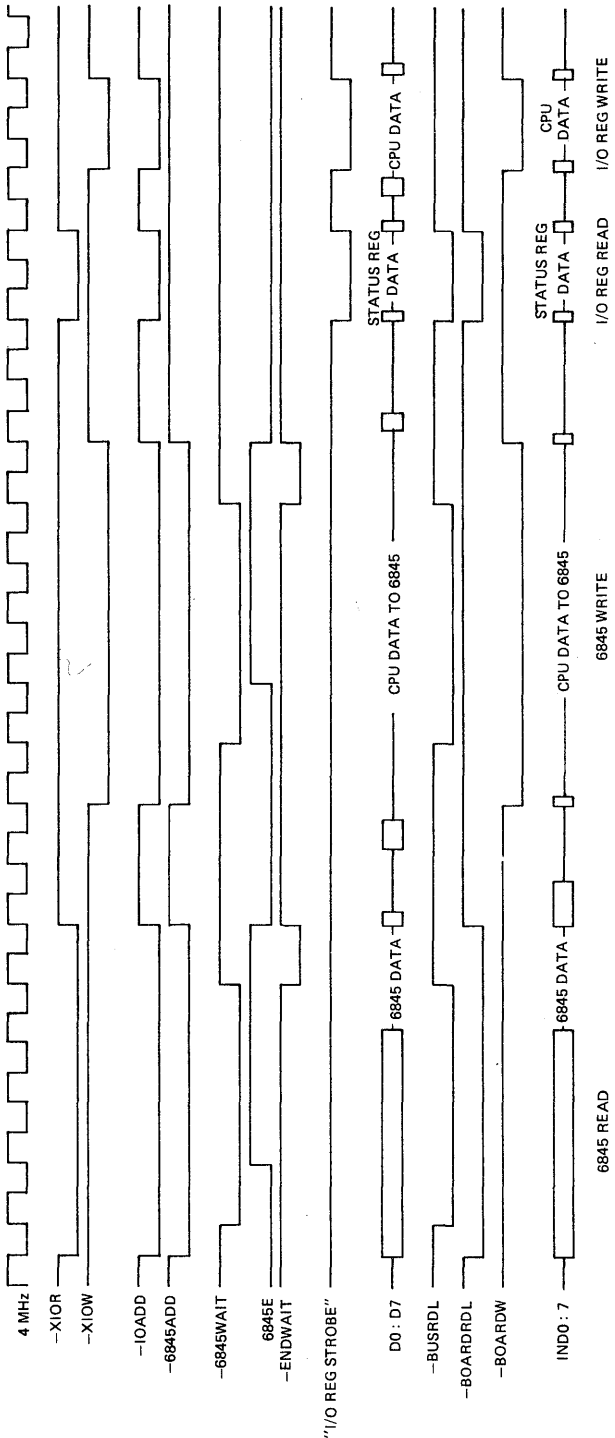


Fig. 5-25 6845 and I/O Registers Read/Write Timing Diagram

6. KEYBOARDS

CONTENTS

PAGE

INTRODUCTION

The Olivetti M24 Personal Computer has two different types of keyboard, keyboard 1 and keyboard 2. The two keyboards have the same circuitry but differ in their layouts. Keyboard 2 also has some added features such as more function keys.

The two keyboards have a permanently attached cable which connect them to a connector at the rear of the M24 System Box. This shielded cable has five wires, one of which is for power supply (+12 Vdc), two for ground, one for keyboard data and one for keyboard clock. Its length can vary from 1.5m to 3.5m and it is coiled like a telephone wire.

The Olivetti M21 Personal Computer keyboard has a different layout from both M24 keyboards but the circuitry is similar to both keyboards. This keyboard has a permanently attached cable similar to the M24 keyboards but it is not detachable from the system. It also serves as a cover to the front of the system while the personal computer is being transported from one place to another.

FUNCTIONAL DESCRIPTION

The following is the functional description for the keyboard circuitry which is valid for the three types of keyboards.

The keyboard uses an Intel 8039 microcomputer to carry out and process the keyboard scanning functions. The Intel 8039 is a single chip microcomputer with 128 bytes internal RAM and 27 I/O lines and is used with a 16K EPROM for self testing capacity. The 8039 is able to treat the keyboard matrix, two LEDs, the 'mouse' and the system communication. It uses a 6MHz clock which is provided by the crystal oscillator connected between pin 2 and pin 3.

Two jumpers are situated on pins 33, 34 of the 8039 chip. These jumpers provide a code to the CPU which identifies the keyboard layout.

The internal machine cycle is 2.5us, while the scanning time is about 10ms. The power-on reset is supplied to the keyboard when the signal KBCLOCK from the system goes low for at least 50ms.

The keys are arranged in a 16 row by 8 column matrix and so the maximum number of keys is 128. Rows are normally clamped at "1" (+5V). They go to "0" (0.4V) one at a time. The decoder 74LS145 scans through the rows while the columns are read 10us after the row output by means of the two decoders LN339. A current of 0.5mA flows through a closed key.

The keyboard has various functions including self testing on power up. This self test program is initiated after a hardware reset. It tests the keyboard firmware, check for stuck keys and "debouncing", that is, ensuring that a key is not considered as having been pressed twice if it is accidentally bounced, and reads the specializing jumpers that define the keyboard layout .

The keyboard recognizes each key by its identification number, called its scan code. These scan codes are numbered from 1 to 103. When a key is pressed, the keyboard sends the relevant scan code to the system unit. When the key is released, the keyboard shows this with a different scan code, which is the normal scan code for that key plus hexadecimal 80.

There are therefore two separate scan codes for each key:

- one for when the key is pressed.
- one for when the key is released.

The keyboard is responsible for keeping track of the amount of time a key is kept depressed, and for generating the repeat key signal. All the keys on the keyboard have this repeat function.

If a given key is held depressed for half a second or longer the keyboard will perform an "auto repeat" function. The code of the depressed key will be transmitted at a 15 Hz rate, continuously, until the key is released. The key code transmitted is the code of the last key held depressed for half a second. The "auto repeat" function is valid for all keys.

The ROM-BIOS routine in the system unit is able to establish the difference between a regular keystroke and a repeat key if necessary, keeping track of the scan codes of the keys which have been released. If two signals that a key has been pressed are received for the same key when there is no signal that the key has been released, this means that the key in question is being kept depressed. This scheme is used by the ROM-BIOS to suppress the repeat function for keys which should not have this function, for example the shift keys.

The keyboard is responsible for an exact indication of the operation being executed, e.g. the pressing or releasing of a key, or a repeat action. The ROM-BIOS is responsible for giving a meaning to all the information sent to it by the keyboard. The keyboard handles the mechanical action of the keys, while the ROM-BIOS programs in the system unit handle all the logic of interpreting the key functions.

If two or more keys are depressed simultaneously the keyboard validates all of them and sends all their codes to the system. The sequence of the code sent is not the real sequence in which the keys are depressed but depends on the scan timing. This rule is valid for all the keys including the SHIFT, CONTROL, ALTERNATE or COMMAND, CAPS LOCK, FUNCTION LOCK or NUMERIC LOCK.

There is a normal key function with no shift action, e.g. the production of lower-case characters. There is also a conventional shift, corresponding to the shift on a normal typewriter, which produces upper-case characters and anything written on the top half of the key.

There are also two special shift keys : ALT - alternate shift, and CTRL - control shift. These work like the ordinary shift keys.

Not all possible combinations of shift keys and ordinary keys are considered acceptable. If a combination which is not considered correct is entered, the ROM-BIOS routine will simply ignore it. These keyboards have four special "toggle" keys, which function as on-off switches in the execution of their particular functions. The keys are :

- Insert
- Caps Lock
- Num Lock
- Scroll Lock

Two of these "toggle" keys, Caps Lock and Num Lock, are part of the keyboard shift mechanism, while the other two, Insert and Scroll Lock, control their own special information.

When the ROM-BIOS receives a scan code from a key other than a shift key, it checks the state of the various shift possibilities and then translates the key into its correct meaning, which might be an ASCII code or a special key code.

Two sets of characters are available in the ROM-BIOS. The first is the ordinary ASCII characters set, which consists of all 256 possible byte codes with the exception of the zero value byte. The codes can be generated on the keyboard either by means of normal keying-in (key A for "A" and so on), or by using the ALT-numeric keys.

Conversely, the special characters are used to indicate the special keys, such as Home, End and the function keys. These special character codes mean that the special keys, such as function keys, can be used without using up any of the 256 ASCII codes.

The keyboards also contain 2 LEDs. On keyboard 1 these LEDs indicate that the CAPS LOCK or the FUNCTION LOCK key has been pressed. On keyboard 2, they indicate that the CAPS LOCK or NUMERIC LOCK key has been pressed.

KEYBOARD SCAN CODES

Key Posn	Scan Code	Key Posn	Scan Code	Key Posn	Scan Code
1	01H	36	24H	71	47H
2	02H	37	25H	72	48H
3	03H	38	26H	73	49H
4	04H	39	27H	74	4AH
5	05H	40	28H	75	4BH
6	06H	41	29H	76	4CH
7	07H	42	2AH	77	4DH
8	08H	43	2BH	78	4EH
9	09H	44	2CH	79	4FH
10	0AH	45	2DH	80	50H
11	0BH	46	2EH	81	51H
12	0CH	47	2FH	82	52H
13	0DH	48	30H	83	53H
14	0EH	49	31H	84	54H
15	0FH	50	32H	85	55H
16	10H	51	33H	86	56H
17	11H	52	34H	87	57H
18	12H	53	35H	88	58H
19	13H	54	36H	89	59H
20	14H	55	37H	90	5AH
21	15H	56	38H	91	5BH
22	15H	57	39H	92	5CH
23	17H	58	3AH	93	5DH
24	18H	59	3BH	94	5EH
25	19H	60	3CH	95	5FH
26	1AH	61	3DH	96	60H
27	1BH	62	3EH	97	61H
28	1CH	63	3FH	98	62H
29	1DH	64	40H	99	63H
30	1EH	65	41H	100	64H
31	1FH	66	42H	101	65H
32	20H	67	43H	102	66H
33	21H	68	44H	103	67H
34	22H	69	45H	104	68H
35	23H	70	46H	-	-

This table shows the key position numbers and the corresponding scan codes in hex. The key position number is shown beside each key in the keyboard layouts shown later.

CHARACTERISTICS

The following table summarizes the characteristics for the M24 keyboards.

Dimensions	Length : 442 mm Height : 30 mm Width : 195 mm
Connector	9-pin "D" Type Cannon Connector
Location	Back of system unit
Cable	1.5m - 3.5m 5 wires, coiled and shielded
Characteristics	1. 4 or 8 degrees of tilt 2. 83 or 103 keys, including 10 or 18 special function keys and 10 numeric cursor control keys 3. Automatic repeat function with a velocity of 10 characters per second 4. Intel 8039 Microprocessor with self-testing capacity 5. Non-generation of the ASCII code

This table shows the pin numbers and specifications for the keyboard interface connector.

Pin	TTL Signals	Signal Level
1	+Keyboard Data	+ 5 Vdc
2	+Keyboard Clock	+ 5 Vdc
3	Ground	0
4	Ground	0
5	+12 Volts	+ 12 Vdc

M24 KEYBOARD 1

This section describes the characteristics of keyboard 1 which differ from those of keyboard 2.

Keyboard 1 has two different positions which can be adjusted to suit the operator; it can be tilted 4 or 8 degrees. The central part of the keyboard is standard, while there are 10 function keys on the left. There are 15 numeric keys on the right-hand side which have a legend for their numeric functions, cursor control, numeric calculations and display editing.

The following list examines the most significant key combinations for this keyboard.

It should be noted that the " CTRL " key, when used in conjunction with other keys, must be kept depressed while the other key is pressed.

1. CTRL SCR LOCK causes the program being executed to be suspended (break key).

2. CTRL NUM LOCK interrupts scrolling on the screen or scrolling and printing; these functions are continued when any key is pressed.

3. CTRL <left arrow> moves the cursor back by one word.

4. CTRL <right arrow> moves the cursor forwards by one word.

5. CTRL END erases the display on the screen from the current cursor position to the end of the line.

6. CTRL PG DN erases the display from the current cursor position to the end of the screen.

7. CTRL PRTSC ensures that the output is sent to the display and to the printer at the same time.

8. CTRL ALT DEL causes a system reset, as if the computer had been turned off and then turned on again.

9. CTRL */PRSTC tells the Personal Computer to print the contents displayed on the screen.

10. ESC (escape) deletes the line the cursor is on.

11. ALT is used to enter any ASCII character code directly from the keyboard.

Figure 6-1 shows the keyboard 1 layout and includes the key position numbers used in the scan code table.

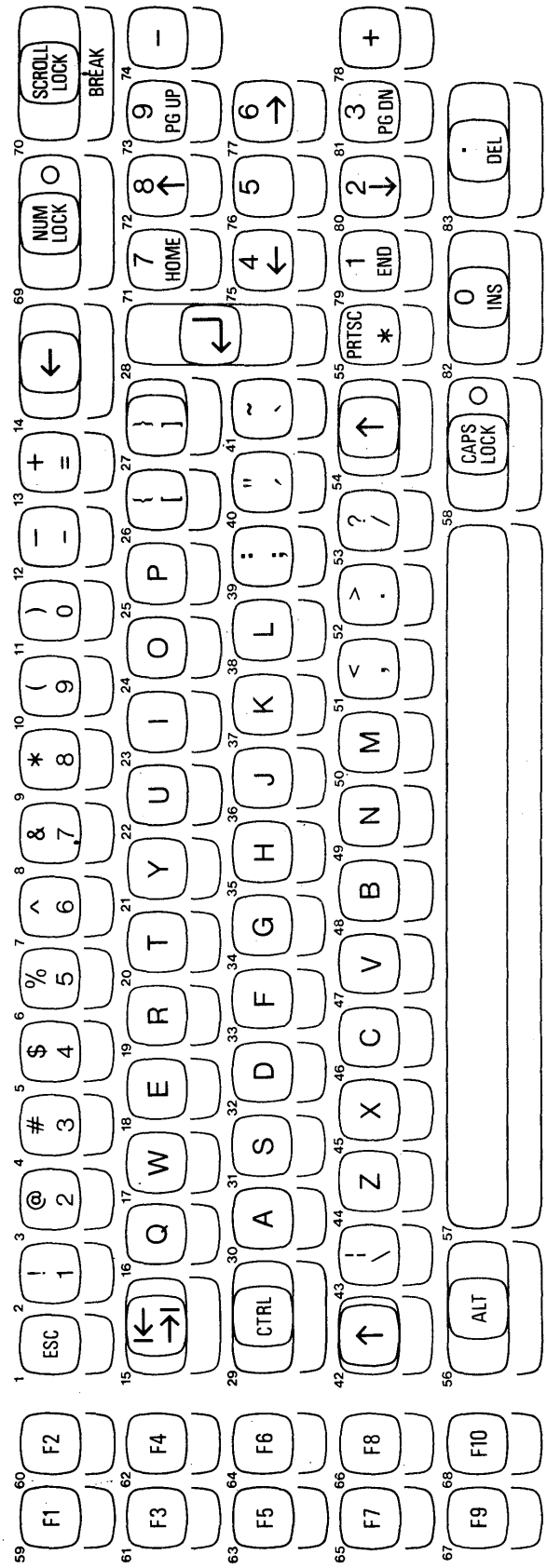


Fig. 6-1 KEYBOARD 1 LAYOUT

M24 KEYBOARD 2

This section describes the characteristics of keyboard 2 which differ from those of keyboard 1.

As can be seen in the following layout, this keyboard contains 103 keys, 18 of which are special function keys. The additional features provided include the possibility of connecting a "mouse" to this keyboard.

A "mouse" is an interactive input device for graphics and test manipulation on an alphanumeric screen. The principle of the mechanical mouse is to encode and transfer to the computer the displacement of the mouse which moves on a smooth surface. The three keys present on the mouse are used to trigger a particular action. Moving the mouse across a table surface rotates a steel rolling pin; the rotation is divided into two perpendicular axes. Two square signals which are out of phase by 90 degrees are produced for each axis, and, depending on the speed at which the mouse is moved, about 8 pulses per millimeter are provided. The pulse generation rate at maximum hand speed is about 1 ms. The standard connector on the mouse is a male 9-pin CANNON sub-miniature connector. The standard cable is an 80 cm. long, 9-wire unshielded cable. The connector pins are:

Pin 1	- +5V	Pin 6	- GND
Pin 2	- Ya	Pin 7	- Middle Switch
Pin 3	- Yb	Pin 8	- Right Switch
Pin 4	- Xa	Pin 9	- Left Switch
Pin 5	- Xb		

Figure 6-2 shows the keyboard 2 layout and includes the key position numbers used in the scan code table.

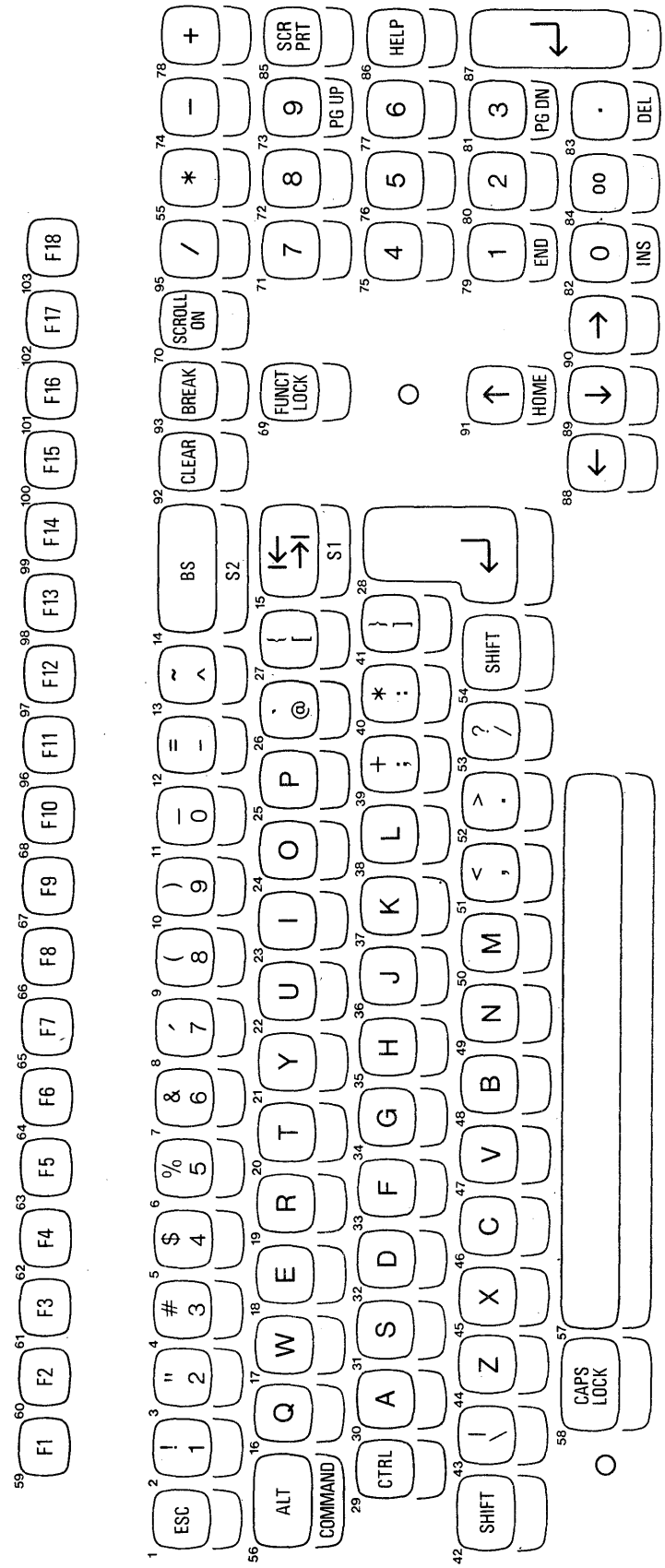


Fig. 6-2 M24 Keyboard 2 Layout

M21 KEYBOARD

The M21 keyboard is an IBM look-a-like keyboard and it's circuitry is the same as the circuitry for the M24 keyboards. The M21 keyboard is similar to the keyboard 1 layout for the M24. The only difference is that the 10 function keys, F1 - F10, are situated in a single row above the main keyboard area. Also, no mouse connector is available on this keyboard.

Figure 6-3 shows the M21 keyboard layout and includes the key position numbers used in the scan code table.

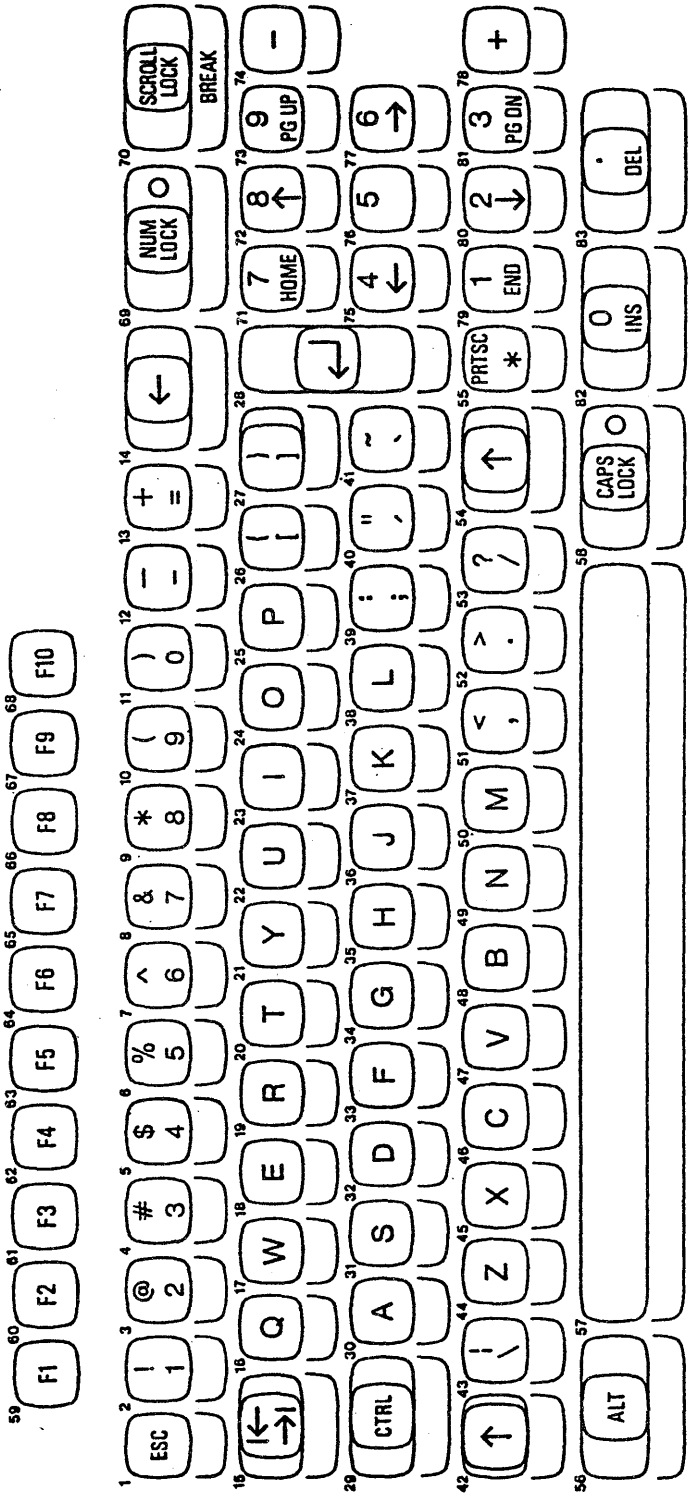


Fig. 6-3 M21 Keyboard Layout

7. POWER SUPPLY

CONTENTS

INTRODUCTION

The Olivetti M24/M21 Personal Computer uses a switching type power supply, with the following electrical input characteristics:

- Single-phase alternating current for the two voltage ranges 100 - 120V and 200 - 240V, adaptable by means of a mains voltage changeover jumper mounted on the printed circuit.

- Permanent driftings of the nominal voltage ±10%, which allows two ranges of correct operation: 90 - 132V and 180 - 264V.

- Nominal operating frequency of between 50 and 60 Hz, with acceptable variations in the order of ±5%.

- Power-up current. A current surge is acceptable in the power-up transient as long as it does not exceed 15 Amperes.

A special filter inserted on the power supply has the function of limiting interference to the mains.
Acoustic noise is < or = 33 dBA.

The general output characteristics can be seen in the following table:

Nominal Volt.	Tolerance	Min Current	Max Current	Max Ripple mVp-p
+ 5V	±5%	6.2 Amperes	16.8 Amperes	50
+12V	±5%	0.9 Amperes	4.5 Amperes	100
+15V	±10%	1 Ampere	1.8 Amperes	150
-12V	±5%	----------	0.25 Amperes	100

The +12VDC can absorb current peaks of 1.4 A caused by the floppy disks. On power-up, this voltage can absorb the current peaks generated when a hard-disk motor is started up, even if they are not within tolerance.
The +5VDC is protected from overvoltage by SCR for values exceeding 6.25V (±0.6V). It is also protected from direct short circuits and from overloads so that the maximum capacity of the power supply is not exceeded. In this case there is a special protection mechanism which, on sensing the +5VDC, turns the power supply off before the current reaches values which are incompatible with the rectification diodes.

The only voltage calibration potentiometer present on the power supply refers mainly to this voltage. Any variation of the +5VDC will obviously also affect the +12VDC.

The +12VDC is protected from short circuits and overloads which exceed the maximum power supplied, and a special protection mechanism ensures that the power supply is turned off before any damage is done.

The -12VDC is regulated by an integrated regulator and is automatically protected from short circuits and overloads. It is also protected from any possible short circuits of the positive voltages, while the +15VDC is only protected from short circuits.

The +15VDC is only protected from short circuits.

MECHANICAL CHARACTERISTICS

The useful dimensions are: 175 x 112 x 112 mm.
The mains switch and the input and output plugs for the power supply to the colour display are all housed in the power supply unit.

The various connectors for the distribution of the supplies within the system are shown in the following figure.

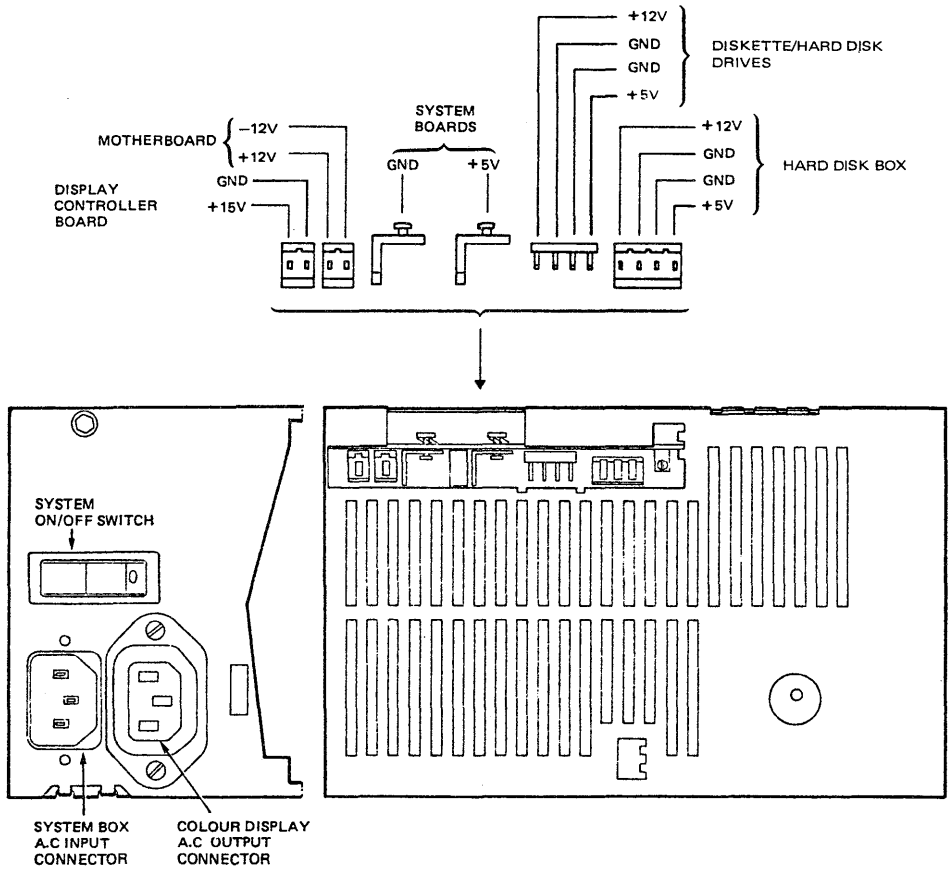


Fig. 7-1 Power Supply Connectors

The power supply uses forced ventilation.
Physically the power supply is composed of two superimposed printed circuit boards plus 6 DC connectors (2 for the motherboard, 1 to connect the mini-floppy disk drives, 1 for the display and 2 to connect the two boards). There are two terminal strips for AC input plus connections for the fan and the ON/OFF switch.

The two printed circuit boards are:

1) Converter Board (lower board)
2) Voltage Regulator Board (upper board)

The converter board, besides accommodating the mains filter, contains the circuitry needed to generate a high frequency (about 25 KHz) alternating current which is then sent to the voltage regulator board.

The second board has the function of transforming and rectifying the voltage sent by the converter, and of ensuring, via a special control circuit, that the required output voltages are sufficiently stable.

FUNCTIONAL DESCRIPTION

The M24/M21 power supply can be defined as a "mains switching converter half- bridge" type.
The main characteristic of this power supply is the regulation of the voltage on the primary winding of the output transformer. This voltage regulation is not obtained by varying the conduction time of the converter's transistors, but by placing a saturable reactor in series with the output transformer.

The AC mains voltage is fed into the power supply terminal strip. The ON/OFF system switch at the rear of the Basic module and the plug for the colour display are connected to this terminal strip.

A voltage selector jumper is present on the lower printed circuit board of the power supply. This jumper selects one of the following voltages:

100 - 120 VAC (jumper present)
200 - 240 VAC (no jumper present)

INPUT CIRCUITRY

A 3.15A 250V fuse is inserted to protect the power supply from short circuits due to power supply failures.
A line filter is also present, which is composed of two LC networks. This reduces the noise created by the power supply itself. Between the bridge rectifier (D141) and the smoothing capacitors (C733) two thermistors (RT02) are inserted. These thermistors have the function of limiting the spurious current peaks created on switching on the ON/OFF switch. The two series smoothing capacitors (C733) double the voltage when the power supply is jumpered for 110 VAC and also act as a convenient halfway voltage reference point.
The output transformer (T438) and the saturable reactor (T365) are in fact inserted between this reference point and the converter's two transistors (Q125).

CONVERTER CIRCUITRY

The converter oscillates because of transformer T366, the magnetic characteristics of which, together with the supply voltage (obtained from both transformer and reactor), causes the two transistors to conduct alternately.

The reactor T365 functions as an ON/OFF switch according to the amount of current sent to its control windings. T365 is constructed so as to provide maximum impedance. No voltage is thus applied to the primary of T438 which in turn does not transfer any power to its secondary.

As the control current increases gradually, the reactor changes state from the previous open condition (maximum impedance) to a closed condition (impedance decreases). Hence all the rectified voltage is present on the transformer T438 which transfers this voltage to the outputs.

The ouput voltage control circuitry then has to provide the reactor with the exact current so that the output voltage is kept at the desired level, increasing the current if the voltage is found to be lower than desired or decreasing it if necessary.

A characteristic of all saturable reactors like T365 is to have the control and load currents related to each other by the number of turnings of the respective windings.
The load current is the current which flows in the T365 itself.

For example, if the ratio of the turnings is 1:1, 2A of control current is needed to have a load current of 2A. This may seem a big limitation since if a favourable turnings ratio is not possible, the control current will always be a significant fraction of the load current (1/2 or 1/3 etc). However, this system has the advantage of automatically limiting the load current to its maximum value.
In fact, by limiting the increase of the control current, the load current is also limited.

On the output side, the voltage is half wave rectified by pairs of diodes, and filtered by a multiple inductance (I100). Besides filtering the voltage, this inductance has the function of connecting the various voltages in such a way as to render them more stable and less prone to load fluctuations.

CONTROL & OVERVOLTAGE CIRCUITRY

The +5VDC, +12VDC and the +15VDC are directly output, as they are within the required stabilization tolerances. For the -12VDC an integrated circuit regulator (VRA8) is needed to stabilize the voltage and to limit the maximum load current to 1 Ampere.

An overvoltage protection mechanism is present on the +5VDC circuitry. This, by means of an SCR (D105), cuts off the +12VDC, which in turn cuts off the +5VDC by means of a diode (D005).
The power supply control circuitry has various functions, the first of which is, naturally, to control the output voltages.

Another function is to interrupt the power output when there is a mains voltage surge in the presence of a short circuit or over current condition at the output side.

The +5VDC and +12VDC output voltages are compared to a reference voltage by means of the differential amplifier circuitry (Q091).
The difference voltage is then amplified by an NPN transistor (Q093) which in turn drives the final control transistor (Q023). This final transistor varies the current associated with the saturable reactor and hence the stabilization of the output voltages is obtained.

An auxiliary voltage proportional to the mains voltage is also taken from the saturable reactor and used for the protection circuits.

In fact, the emitter of the PNP transistor (Q091) is connected to this voltage and a fraction of the +12V output is fed on the base of the same transistor.

In normal working conditions, the transistor (Q091) is open and hence the base is at a higher potential than the emitter. But if the mains supply voltage increases, or if the output DC voltages decrease, the transistor starts conducting and in turn saturates the NPN transistor (Q093) which opens the final control transistor (Q023).

Cutting off the output voltages means that the base of the NPN PNP transistor (Q091) is now almost at ground potential and hence the auxiliary voltage must decrease to zero to get out of saturation. The only way to achieve this is to switch off the power supply.

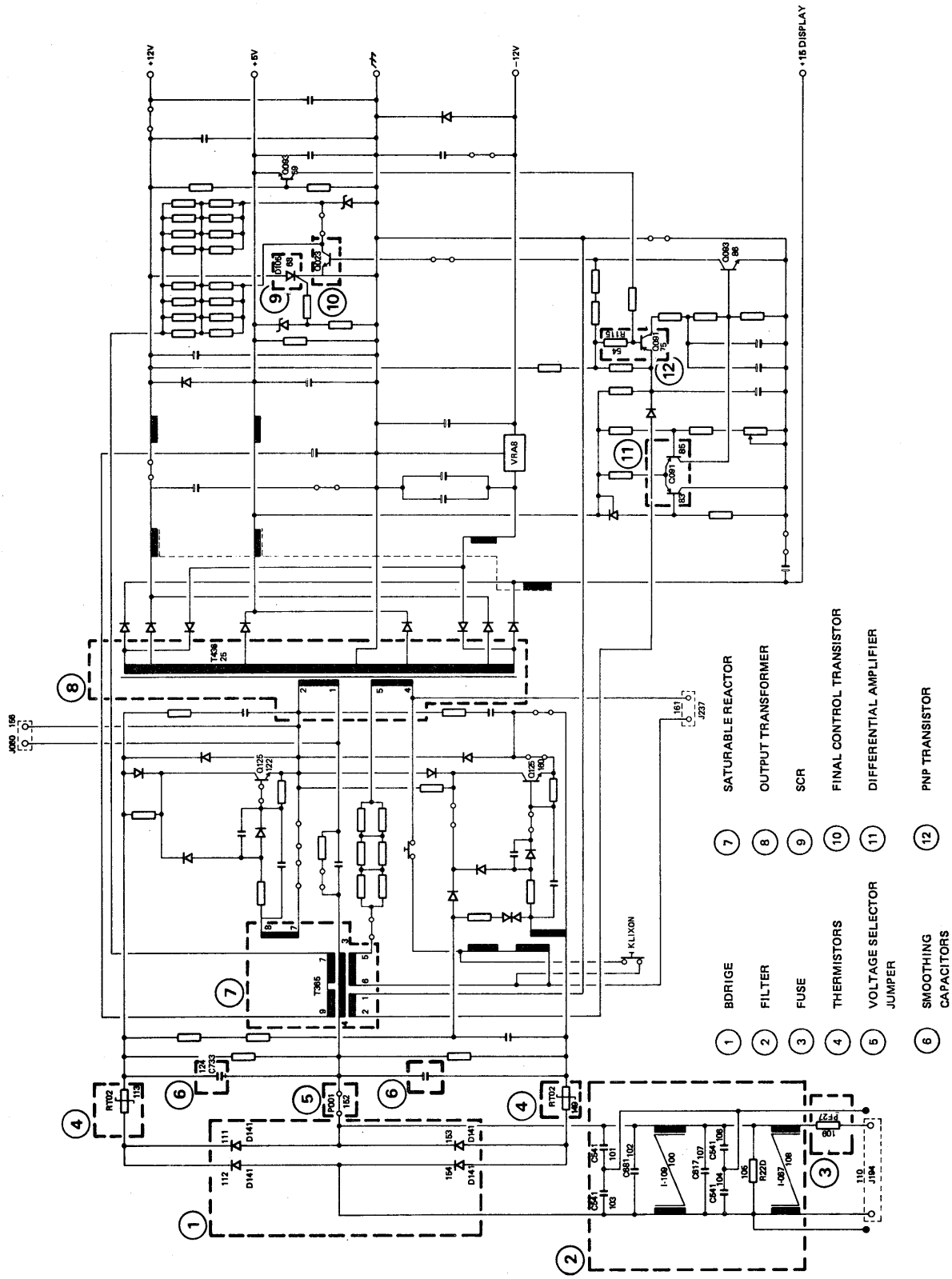


Fig. 7-2 Power Supply Circuit Diagram

8. APB Z8000 BOARD (APB 2481)

CONTENTS

OVERVIEW

The APB Z8000 option is a plug-in board which allows the system to execute software written for a Z8001 microprocessor. The purpose of this board is to allow the PC system to work in a PCOS operating system environment used by the M20 system.

It plugs into the system bus converter dual row expansion slots and is based on a 4 MHz Z8001 microprocessor with on board boot ROMs. It also contains an RS232 serial interface and connector which can be used together or instead of the RS232 connector present on the motherboard. An address translation PROM is used to change the logical addresses generated by the Z8001 processor to the physical memory addresses present on the motherboard, the memory extension board and the display controller boards.

This board permits the system to run all the software written for the M20 system and allows the personal computer to be used in an M20 local area network configuration. It provides monochromatic graphics with a resolution of 512 x 256 pixels and alphanumerics with a resolution of 64 x 16 and 80 x 25. It can also provide 4 and 8 colour graphics if the display controller options board is present.

GENERAL DESCRIPTION

The APB Z8000 board, if present, is activated via software by the 8086 processor present on the system motherboard, since the APB Z8000 board is in the deactivated state following power-on or reset. The following figure is a flow chart illustrating what takes place when the system is switched on with an APB Z8000 board plugged into the bus converter board.

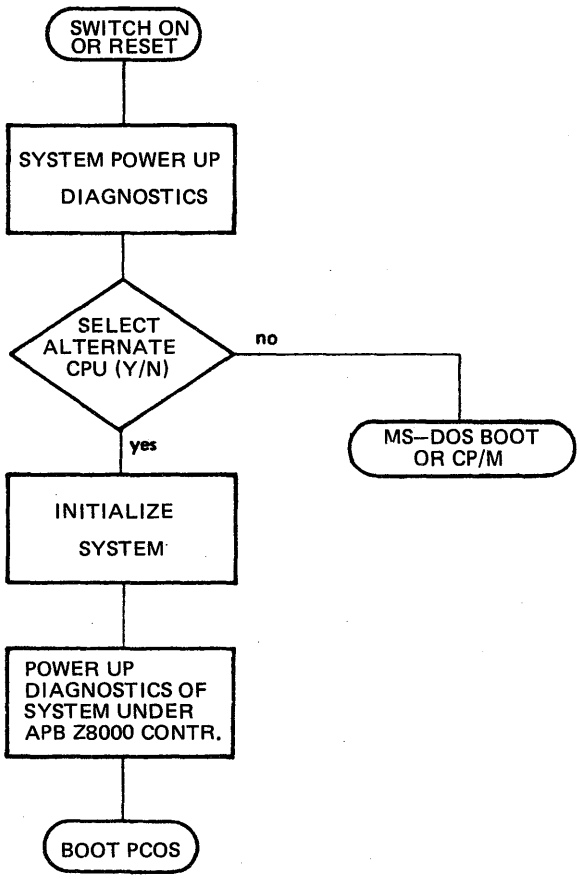


Fig. 8-1 APB Z8000 Booting Flow Chart

After power-on or reset the system performs the power-on diagnostics assuring proper operation of the system. The operator is then asked whether he wants the system to run under the control of the 8086 CPU present on the motherboard or the Z8001 alternate processor present on the APB Z8000 board.

If the operator selects the alternate CPU, the system is then initialized and a series of tests on the system are performed. If these tests are successful, the PCOS operating system is booted from the disk drive and the system is now ready to run in a PCOS environment.

FUNCTIONAL DESCRIPTION

The following figure is a functional block diagram of the APB Z8000 board. The principal circuits shown in this block diagram are:

- Z8001 CPU
- Address Mapping PROM and Buffers
- Read Only Memory
- Bus Handshaking Logic
- I/O Ports and Devices
- RS 232 Serial Interface
- Interrupt and Wait Logic

The description that follows gives an explanation of these circuits.

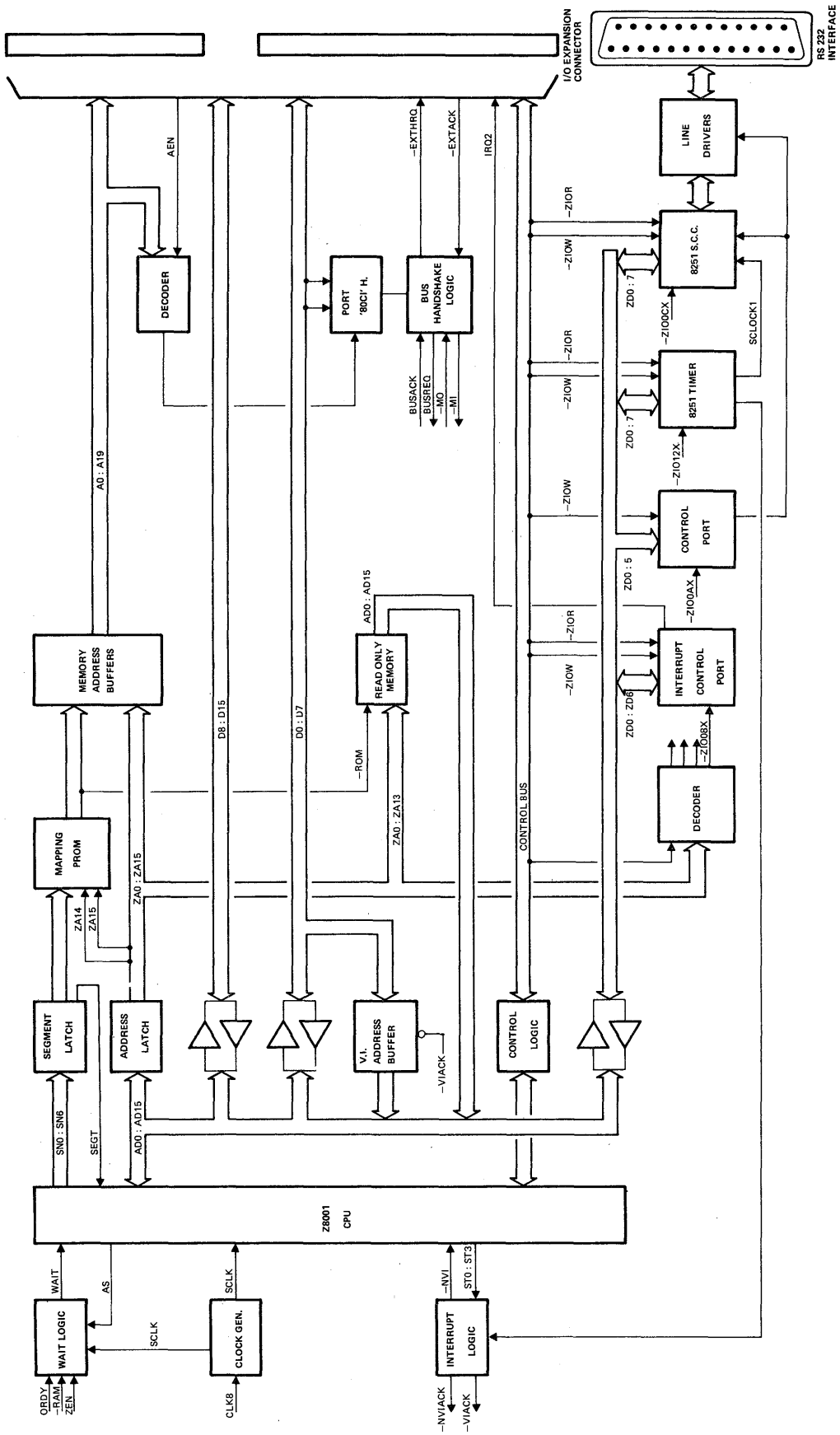


Fig. 8-2 APB Z8000 Block Diagram

BUS HANDSHAKING

Normally the 8086 processor is in control of the system bus. When PCOS is required, the Z8000 board must come into action and the Z8001 processor must gain control of the system bus. This is done in the following manner:

During normal operation and whenever the system is reset, the Z8000 board is forced into a permanent reset state. The 8086 processor on the motherboard starts the process of detecting the existence of the Z8000 board by writing hex '01' to '80C1' hex. This means that A0, A6, A7 and A15 are all high. A15 high, A4, A8 and A9 low enable the multiplexer 74LS138 and the Y6 output, -IOCX, goes low since A6 and A7 are high and A5 is low.

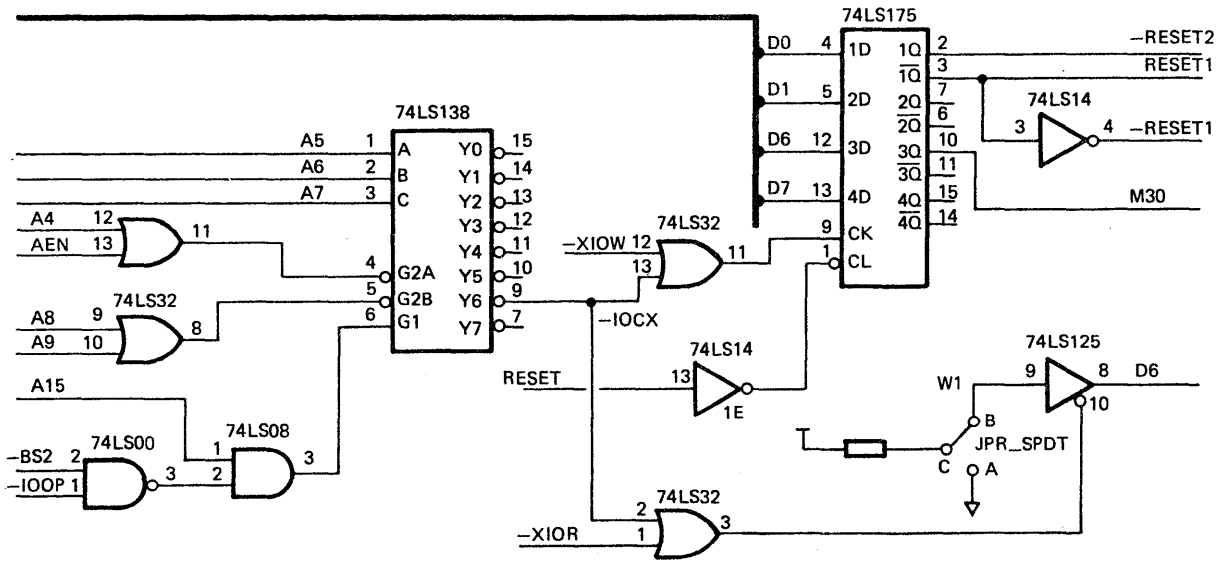


Fig. 8-3 I/O Port '80C1' Circuitry

-IOCX going low activates the tristate drivers 74LS125 and thus causes the signal -16BCH low. This signals to the motherboard that a 16 bit board is in use.

While the 8086 is writing hex '01' to port '80C1', the signal -XIOW is low. At the end of this I/O write, -XIOW goes high clocking the values of D0 and D6 into the quad flipflops 74LS175. Since D0 is high, the signal -RESET2 goes high and RESET1 goes low.

The signals -RESET2 and RESET1 are the board select signals and release all the board circuitry from the reset state. The signal RESET1 releases the Z8001 processor from the reset state which then fetches the initialization codes from the on board ROMs and tries to gain control of the system bus by executing a MREQ (Multi Micro Request) instruction. The 8086 allows enough time for the Z8001 to request the bus. If the Z8001 fails to respond within the time limit, it is assumed to be absent.

When the Z8001 gains control of the bus, the Z8001 processor writes hex '20' on the ZD bus and '00AX' on the ZA bus. Thus the signal -ZI00AX goes low and ZD5 goes high. This generates the signal ZEN high and -ZEN low, enabling memory access to the system memory. It then reads port '80C1' and since the contents of this port is '01', it implies that the 8086 is querying about the Z8001's presence. So the Z8001 reads memory location '0' and if its contents is '0', the Z8001 writes a '1' to that location and then writes '00' to port '80C1'. This causes -RESET2 to go low and RESET1 to go high which then reset the Z8001 processor and its associated circuitry.

As the 8086 regains control of the bus, it goes to memory location '0' to find out if the Z8001 identified itself. Then the 8086 asks the user if the Z8001 board should be activated. If so, it sets the alternate processor bit, ALT, in the control port '8086' and writes '0F' to memory location '0' and '01' to port '80C1 as before. This means that the 8086 gives up the system bus permanently and goes into a permanent reset state.

The Z8001 now executes a MREQ instruction to get the bus and writes hex '20' to port '0A' and reads memory location '0' and finds '0F'. Thus, now, the Z8001 is in control of the system bus and holds it till a hardware reset is performed.

BUS HANDSHAKING TIMING

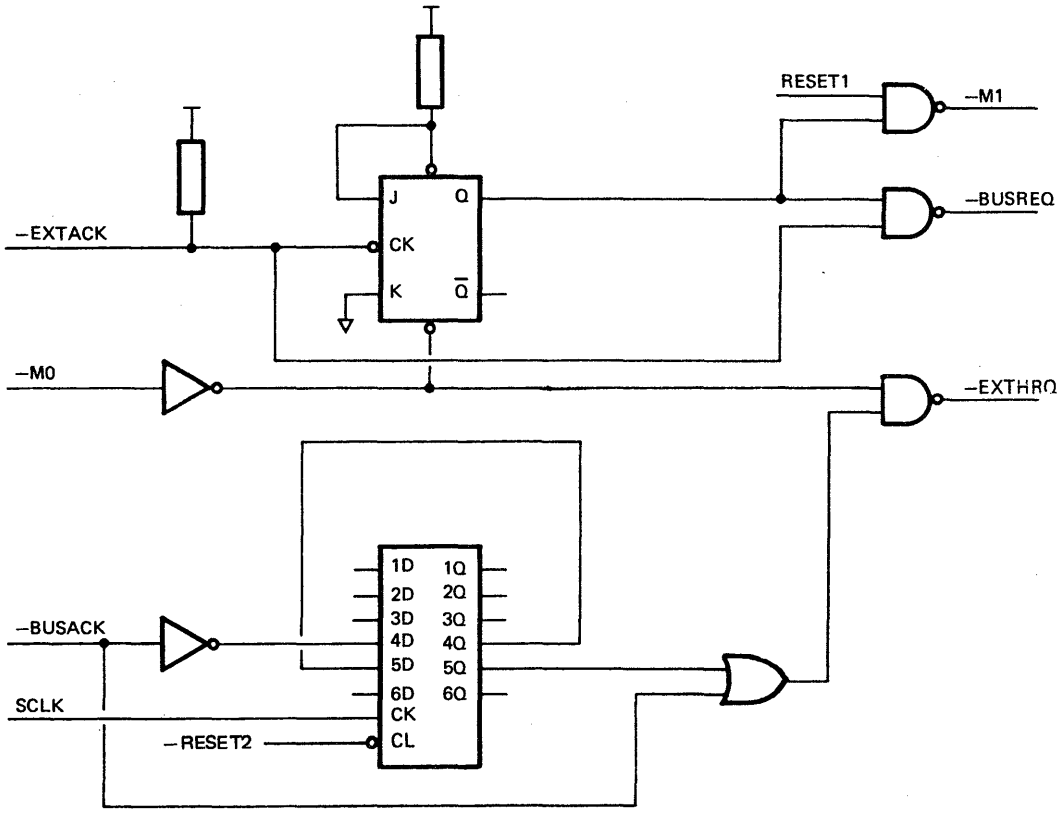


Fig. 8-4 Bus Handshaking Circuitry

The signal RESET1 going low, makes the signal -MI to go high. This indicates that a MREQ instruction is to be performed. Thus the Z8001 forces the -M0 pin low which causes the external hold request -EXTHRQ to go active low. This signal goes to the bus arbiter logic on the motherboard which then responds with external acknowledge signal -EXTACK low.

The ALT bit in the bus arbiter logic on the motherboard is set by D6 when the -IOCX is active or when -IOR is active. With the ALT bit set and -EXTACK low the 8086 processor is put into a wait state till a hardware reset is performed. -EXTACK low clocks the flipflop 74LS112 and forces -MI low causing the Z8001 to gain control of the system bus.

If the Z8001 is in control of the system bus and the DMA controller requires the use of the bus, the DMA controller issues a DMA hold request DMAHRQ which causes the signal -EXTACK to go high. This signal comes on to the Z8000 board and forces the signal -BUSREQ low which then goes to the Z8001 processor which replies with -BUSACK. This signal low causes -EXTHRQ to go high and thus the DMA controller takes control of the system bus.

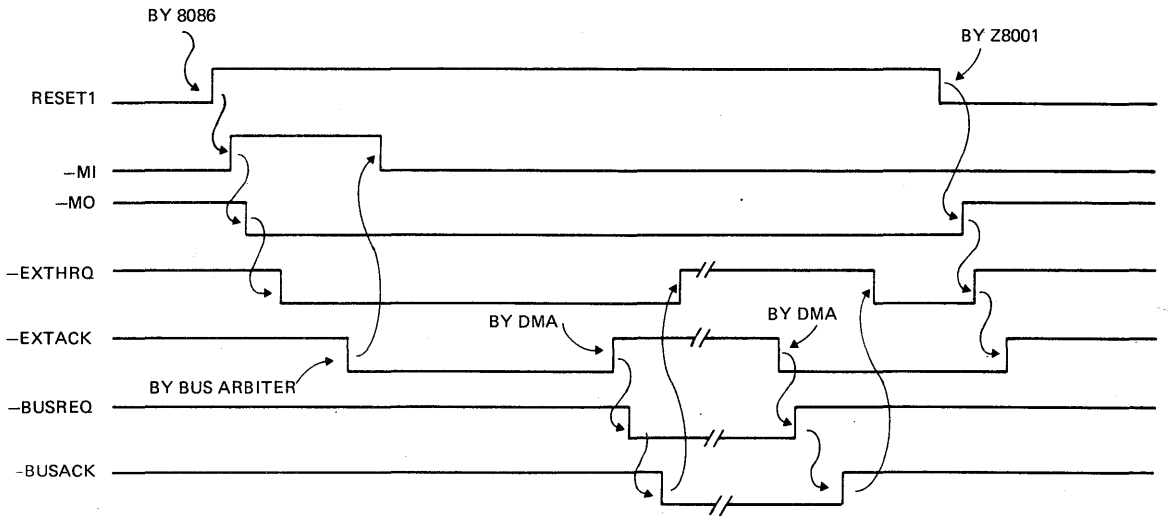


Fig. 8-5 Timing Diagram

When the DMA controller finishes with the bus, it forces the signal -EXTACK low which causes the signals -BUSREQ and -BUSACK to go high and -EXTHRQ to go low.

Now the bus is again under control of the Z8001 processor. If the Z8001 finishes with the bus and wants to release it to the 8086 processor, the -RESET2 signal goes low and RESET1 goes high resetting the Z8001 and causing the -MO output to go high. This, in turn, forces -EXTHRQ and -EXTACK to go high and thus the 8086 will again take control of the system bus.

Z8001 PROCESSOR

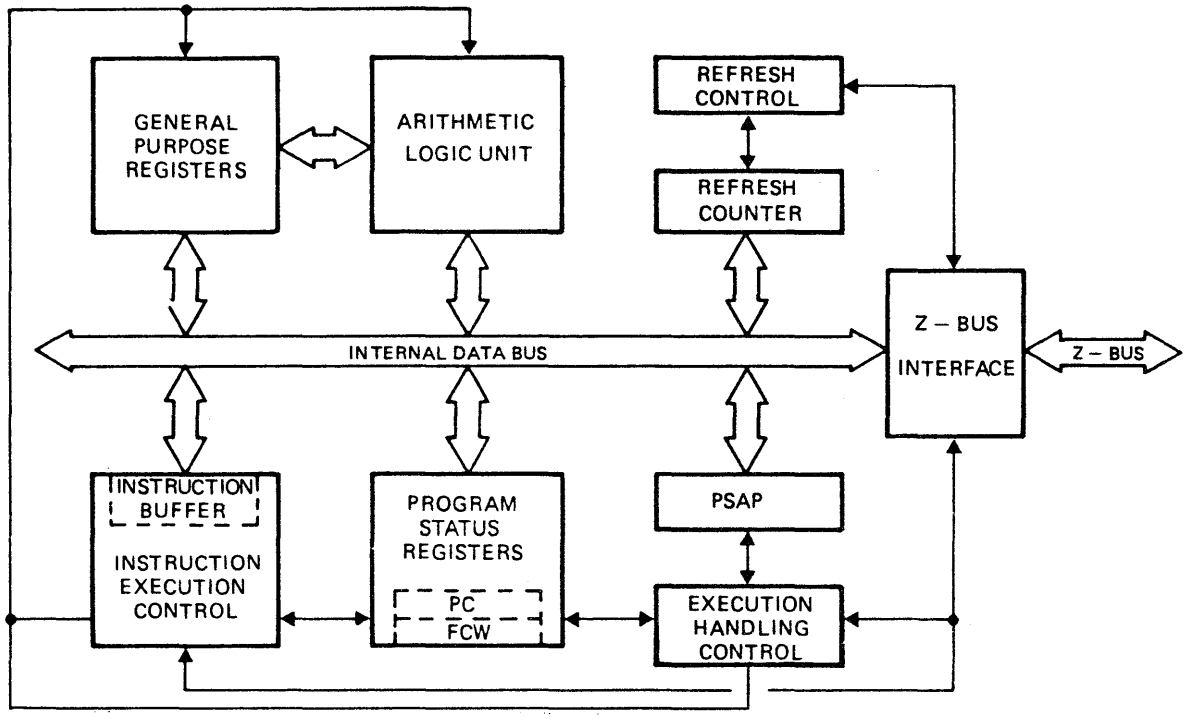


Fig. 8-6 Z8001 Functional Block Diagram

The Z8001 processor is the primary element of the APB Z8000 board. Figure 8-6 is a block diagram that shows the major elements of the Z8001 CPU, namely:

- A 16 bit internal data bus , which is used to move addresses and data within the CPU.
- A Z-Bus interface, which controls the interaction of the CPU with the outside circuitry.
- A set of 16 general purpose registers, which is used to contain addresses and data.
- Four special purpose registers, which control the CPU operation.
- An Arithmetic and Logic Unit, which is used for manipulating data and generating addresses.
- An instruction execution control, which fetches and executes Z8001 instructions.
- An exception handling control, which processes interrupts and traps.
- A refresh control, which generates memory refresh cycles.

Z8001 PIN FUNCTIONS

The Z8001 pins can be grouped into categories according to their function.

Transaction Pins - These signals provide timing and data transfer for bus transactions.

AD0-AD15 Address/Data (output, active high, 3-state): These multiplexed address/data lines carry the input/output addresses, the offset portion of memory addresses and data during bus transactions.

SN0-SN6 Segment Number (output, active high, 3-state): These lines carry the encoded segment number of the memory address.

ST0-ST3 Status (output, active high, 3-state): These lines indicate the kind of transaction occurring on the bus and additional information about the transaction.

ST3	ST2	ST1	ST0	TRANSACTION
0	0	0	0	Internal Operation
0	0	0	1	Memory Refresh
0	0	1	0	Standard I/O
0	0	1	1	Special I/O
0	1	0	0	Segment Trap Acknowledge
0	1	0	1	Non-maskable interrupt Acknowledge
0	1	1	0	Non-vectored interrupt Acknowledge
0	1	1	1	Vectored interrupt Acknowledge
1	0	0	0	Memory data request
1	0	0	1	Memory stack request
1	0	1	0	Reserved
1	0	1	1	Reserved
1	1	0	0	Program reference, nth word
1	1	0	1	Instruction fetch, 1st word
1	1	1	0	Reserved
1	1	1	1	Reserved

B/-W Byte/Word (output, low = word, 3-state): This line indicates whether a byte or word of data is to be transmitted during a transaction.

-WAIT (input, active low): A low on this line indicates that the responding device needs more time to complete a transaction.

-MREQ Memory Request (output, active high, 3-state): A falling edge on this line indicates that the address/data bus is holding a memory address.

-AS Address Strobe (output, active low, 3-state): The rising edge of -AS indicates the beginning of a transaction and shows that the address, ST0-ST3, N/-S, R/-W and B/-W signals are valid.

-DS Data Strobe (output, active low, 3-state): This line provides timing for data movement to or from the CPU.

R/-W Read/Write (output, low = write, 3-state): This line determines the direction of data transfer for memory input/output transactions. For memory read R/-W = high; for memory write R/-W = low. For input/output transactions the R/-W line indicates the direction of data transfer. Peripheral to CPU: Read R/-W = high; CPU to peripheral Write R/-W = low.

N/-S Normal/System Mode (output, low = system mode, 3-state): In system mode, all instructions can be executed and all CPU registers are accessed. This mode is intended for use by programs performing operating system functions. In normal mode some instructions may not be executed and the control registers of the CPU are inaccessible. In general this mode of operation is intended for use by application programs.

Bus Control Pins - These pins carry signals for requesting and obtaining control of the bus from the CPU.

-BUSREQ Bus Request (input, active low): A low on this line indicates that a bus requester has obtained or is trying to obtain control of the bus.

-BUSACK Bus Acknowledge (output, active low): A low on this line indicates that the CPU has relinquished control of the bus in response to a bus request.

Interrupt and Trap Pins - These pins convey interrupt and external trap requests to the CPU.

-NMI Non-maskable Interrupt (input, edge activated): A high to low transition on -NMI requests a non-maskable interrupt. The -NMI interrupt has the highest priority of the three types of interrupts. NMI is reserved for events that require immediate attention.

-NVI Non-vectored Interrupt (input, active low): A low on this line requests a non-vectored interrupt. A non-vectored interrupt relies on the system software to determine its cause.

-VI Vectored Interrupt (input, active low): A low on this line requests a vectored interrupt. A vectored interrupt causes 8 bits of the vector output by the interrupting device to be used to select a particular interrupt service procedure to which the program branches.

-SEGT Segment Trap (input, active low): A low on this line requests a segment trap. Generated by addressing a non-existent segment.

Multi Micro Pins - These lines form a resource-request daisy chain that allows one CPU in a multi-microprocessor system to access a shared resource.

-MI Multi-Micro In (input, active low): This input is used to sample the state of the resource request lines.

-MO Multi-Micro Out (output, active low): This line is used by the CPU to make resource requests.

CPU Control Pins - These pins carry signals which control the overall operation of the CPU.

-STOP (input, active low): This line is used to suspend CPU operation during the fetch of the first word of an instruction.

-RESET (input, active low): A low on this line resets the CPU.

PROGRAMMABLE READ ONLY MEMORY

The Z8000 board contains 16K bytes of resident ROMs. These ROMs serve as memory storage for the power-on bootstrap and power-on diagnostic routines.

The ROMs are enabled by the signal -ROM going low as described in the address translation section. They are addressed by the Z8001 data bus ZA1-ZA13. The eight data outputs of the ROMs go on to the internal data bus. The ROMs are only accessed by the Z8001 local bus and does not involve the system bus. A ROM access takes only 4 processor cycles or 1 usec.

RANDOM ACCESS MEMORY

The APB Z8000 board uses the random access memory present in the system as system memory. It is able to address up to 512K bytes of RAM present on the motherboard and on the memory expansion boards. It can also address another 128K bytes of RAM resident on the display controller boards and used as a display buffer. 32K bytes are always present on the the Indigenous display controller while the other 96K bytes are present on the optional display controller board.

ADDRESS SPACES

Programs and data may be in the main memory of the system or in the peripheral devices. In either case, the location of the information must be specified by an address before the information can be accessed. A set of these addresses is called an address space. The Z8001 supports two different types of addresses and thus two categories of address space.

Memory Addresses - which specify locations in main memory. This space is subcategorized into:

> Instruction space (normal or system mode). These spaces typically address memory that contains user programs (normal mode) or system programs (system).
>
> Data space (normal or system mode). These spaces may be used to address the data that user or system programs operate on.
>
> Stack space (normal or system mode). These spaces may be used to address the system and normal program stacks.

Input/Output Addresses - which specify the ports through which peripheral devices are accessed.

SEGMENTED MEMORY ADDRESS

Segmentation is a means of partitioning memory into segments so that a variety of useful functions may be implemented including protection mechanisms that prevent a user from referencing data belonging to others, attempting to modify read-only data or overflowing a stack. This board uses the Z8001 processor with segmented address capability. Four segment lines and 16 address lines are used to address a total of 512 KB of RAM and ROM. Each segment can hold a maximum of 64 KB.

The Z8001 issues the encoded segment number of the memory address on the segment number lines SN0-SN6. The address strobe AS latches the signals SN0-SN3 onto the ZSN0-ZSN3 lines which are then used to address the address translation PROM.

The segment lines SN4-SN6 should always be low but if a higher segment number is generated one of the SN4-SN6 lines goes high and causes the -SEGT pin of the Z8001 processor to go low. The -SEGT input low signals to the processor that a non-existent segment has been addressed.

The following is a description of the segments as used on the M24 system:

Segment 0: This is a general purpose segment containing both data and program space with a few locations reserved for special purposes. The data area is reserved for the PCOS and Boot ROM disk driver.

Segment 2: This segment is used for several purposes. It contains the system stack, an area for BASIC, several RAM blocks, a non-overlayed RAM area, and some user memory.

Segment 3: This segment contains the display RAM. The physical size of this segment depends on the display configuration. The physical offsets available are 0000 through BFFF for an 8 colour system, 0000 through 7FFF for a 4 colour system, and 0000 through 3FFF for a monochrome system.

Segment 5: This segment is another general purpose segment containing both PCOS data and codes. It contains the bootstrap codes and ROM debugger and some ROM routines. It is physically located in the spare 16K block of display RAM. Since access to this RAM is slower than access to the normal RAM, no time dependent code or data should be placed into this area.

Segment 6: This segment contains the general PCOS data and code areas as well as an area reserved for BASIC. The area 0000 - 7FFF is reserved for BASIC while the PCOS code and data area is limited in the 8000 - BFFF space.

Segment 9, 10: These two segments are dedicated for user memory space if the expansion memory exists. The physical offsets of these segments are from 8000 - FFFF.

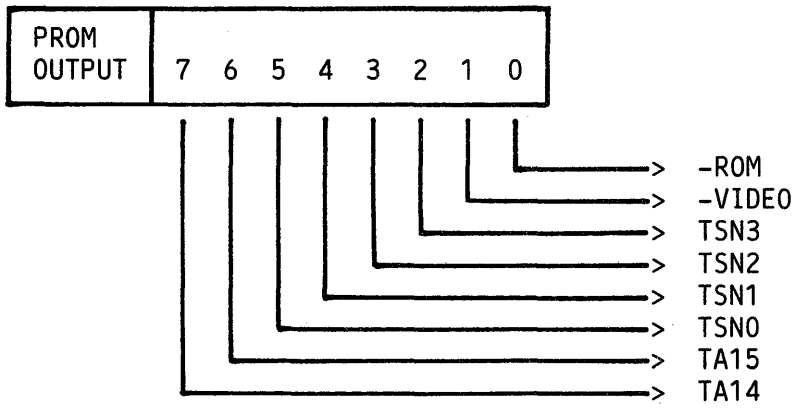
Segment 11-15: All these segments are present on the expansion memory and thus PCOS must always check if this memory exits during initialization. They all range from 0000 - FFFF and are used as user memory space.

ADDRESS TRANSLATION

The address translation/mapping or relationship between the logical memory addresses as viewed by the Z8001 and the physical memory elements as addressed by the system hardware is accomplished through a 256 byte address translation PROM.

The PROM address inputs consist of two Z8001 address lines (ZA14, ZA15), one Z8001 status line (ST2), four Z8001 segment lines (SN0-SN3) and two control lines (M30, -MEMOP). The PROM is enabled when the signal -MEMOP is low meaning that a memory access operation is to be performed.

The PROM outputs consist of two translated address lines (TA14, TA15), four translated segment lines (TSN0-TSN3), and the two device select lines -VIDEO and -ROM.



The signal -ROM low, selects and enables the ROMs present on the Z8000 board. The address bits for these ROMs consist of the Z8001 address lines ZA1-ZA13.

The signal -ROM high enables the multiplexer 74LS139. If the PROM output -VIDEO is low, during a memory operation, the multiplexer output 1Y0 is selected and thus the signals -BUFB and -BUFC are put low. These signals enable the 74LS244 address buffers B and C and activate the tristate driver 74LS125.

Whenever the -VIDEO signal is low, the PROM outputs TSN3, TSN2, TSN1 and TSN0, are always as shown below.

TSN3	TSN2	TSN1	TSN0	
A19	A18	A17	A16	
1	0	1	x	A0000-BFFFF

This means that with these conditions the space addressed is A0000-BFFFF which is the memory space reserved for the display buffer.

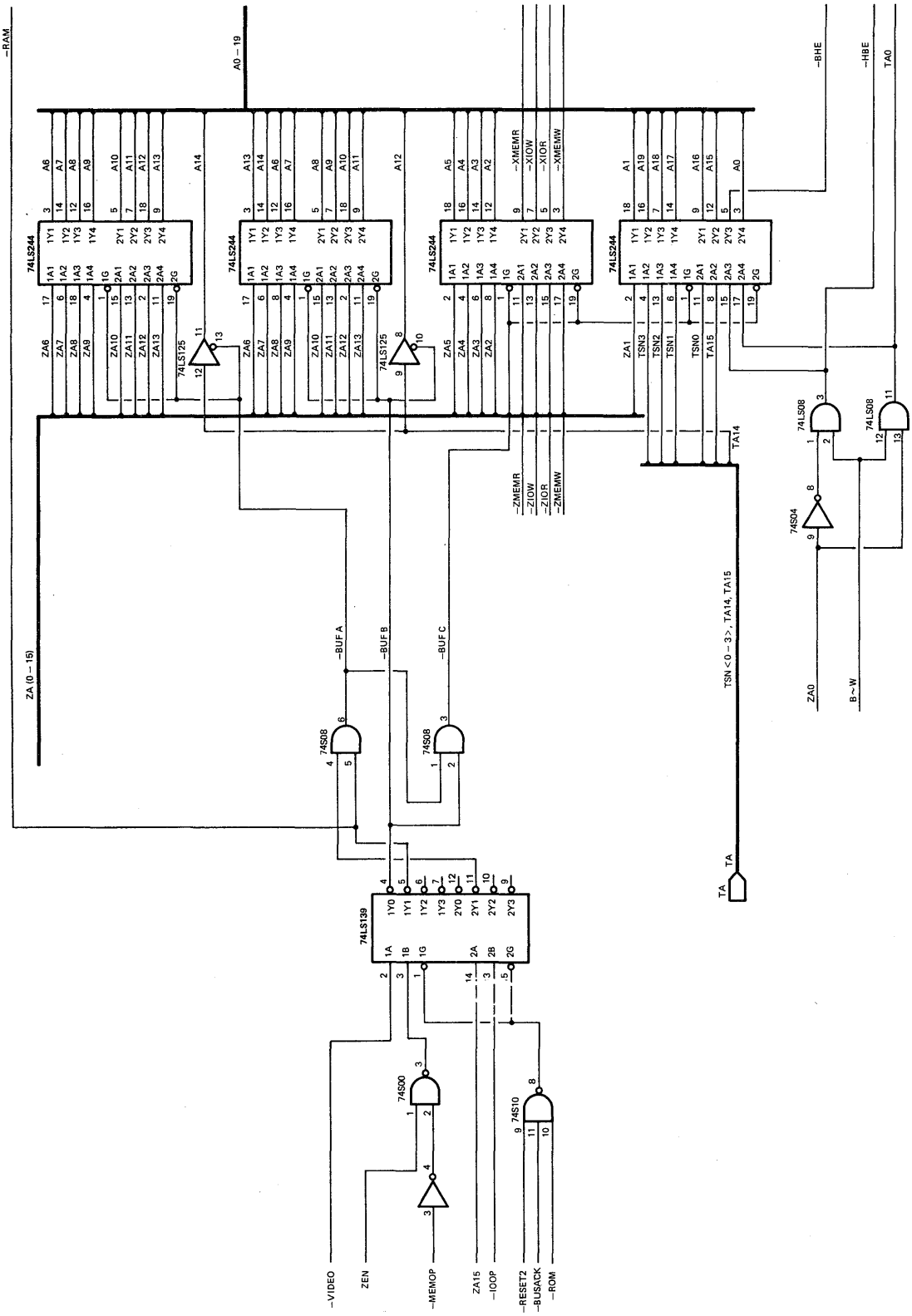


Fig. 8-7 Address Translation Circuitry

Since the indigenous display controller only supports the monochrome display unit in PCOS mode, logical segment 3 offset 0000 to 3FFF is used as display buffer. The rest of the logical segment 3 is used as the video display buffer for colour systems when the display options board is present.

Segment	Offset	Bit Plane
3	0000-3FFF	0
3	4000-7FFF	1
3	8000-BFFF	2
3	C000-FFFF	3

When the signal -VIDEO is high, with -ROM high, it selects the 74LS139 multiplexer output 1Y1, thus putting the signal -RAM low. It also puts the signals -BUFA and -BUFC low which then select the 74LS244 address buffers. This allows the Z8001 to address the RAM present on the motherboard and the memory expansion boards. If TSN2 and TSN3 are both low, then A18 and A19 are both low and thus the first 256 KB of memory on the motherboard can be addressed.

TSN3	TSN2	TSN1	TSN0	
A19	A18	A17	A16	
0	0	x	x	00000-3FFFF

If TSN3 is low and TSN2 is high, then A19 is low and A18 is high. This will allow the Z8001 to address memory locations between 40000-7FFFF.

TSN3	TSN2	TSN1	TSN0	
A19	A18	A17	A16	
0	1	x	x	40000-7FFFF

Logical to Physical Memory Map

CODE SEGMENTS

3		7	V3	R3	Vx		R3	9	11	9	15	19	23	27	31
2	3	6	V2	R2	R3	3	R2	8	10	8	14	18	22	26	30
1	2	5	V1	R1	R2	2	R1	2	6	2	13	17	21	25	29
0	1	4	V0	R0	R1	1	R0	1	5	1	12	16	20	24	28
CS0	CS1	CS2	CS3	CS4	CS5	CS6	CS7	CS8	CS9	CS10	CS11	CS12	CS13	CS14	CS15

DATA/STACK SEGMENTS

3	7	7	V3	R3	Vx		R3	11	11	9	15	19	23	27	31
2	6	6	V2	R2	3	3	R2	10	10	8	14	18	22	26	30
1	5	5	V1	R1	2	2	R1	6	6	2	13	17	21	25	29
0	4	4	V0	R0	1	1	R0	5	5	1	12	16	20	24	28
DS0	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9	DS10	DS11	DS12	DS13	DS14	DS15

Each block in the tables above represents 16 KB of logical memory space in the Z8001 logical memory space. The contents of each box represents the physical mapping of that logical address space as defined below:

R0-R3 are the Z8001 ROM memory.

V0-V3 indicates the display bit planes (V0 = Bit plane 0, V1 = Bit plane 1, etc.).

Vx indicates extra memory which can be used as system memory; the memory access is slower than regular system memory.

The numbers 0-31 indicate 16 KB blocks of normal system memory with the first physical block being 0.

Memory Address Translation Map

CODE SEGMENTS

C3	FF	E3	35	C2	F5	FF	C2	93	D3	93	F3	CB	EB	DB	FB
43	C3	63	55	42	C2	C3	42	13	53	13	73	4B	6B	5B	7B
83	43	A3	15	82	42	43	82	43	63	43	B3	8B	AB	9B	BB
03	83	23	75	02	82	83	02	83	A3	83	33	0B	2B	1B	3B
CS0	CS1	CS2	CS3	CS4	CS5	CS6	CS7	CS8	CS9	CS10	CS11	CS12	CS13	CS14	CS15

DATA/STACK SEGMENTS

C3	E3	E3	35	C2	F5	FF	C2	D3	D3	93	F3	CB	EB	DB	FB
43	63	63	55	42	C3	C3	42	53	53	13	73	4B	6B	5B	7B
83	A3	A3	15	82	43	43	82	63	63	43	B3	8B	AB	9B	BB
03	23	23	75	02	83	83	02	A3	A3	83	33	0B	2B	1B	3B
DS0	DS1	DS2	DS3	DS4	DS5	DS6	DS7	DS8	DS9	DS10	DS11	DS12	DS13	DS14	DS15

The tables above give the actual contents of the address translation PROM.

Z8001 Logical Block Number To 8086 Physical Address Look Up Table

Z8000 Logical Block	8086 Physical Address (A19-A0)
0	00000 - 03FFF
1	04000 - 07FFF
2	08000 - 0BFFF
3	0C000 - 0FFFF
4	10000 - 13FFF
5	14000 - 17FFF
6	18000 - 1BFFF
7	1C000 - 1FFFF
8	20000 - 23000
9	24000 - 27000
10	28000 - 2B000
11	2C000 - 2FFFF
12	30000 - 33FFF
13	34000 - 37FFF
14	38000 - 3BFFF
15	3C000 - 3FFFF
16	40000 - 43FFF
17	44000 - 47FFF
18	48000 - 4BFFF
19	4C000 - 4FFFF
20	50000 - 53FFF
21	54000 - 57FFF
22	58000 - 5BFFF
23	5C000 - 5FFFF
24	60000 - 63FFF
25	64000 - 67FFF
27	6C000 - 6FFFF
28	70000 - 73FFF
29	74000 - 77FFF
30	78000 - 7BFFF
31	7C000 - 7FFFF
R0	DOES NOT EXIST
R1	DOES NOT EXIST
R2	DOES NOT EXIST
R3	DOES NOT EXIST
V0	B8000 - BBFFF
V1	A0000 - A3FFF
V2	A8000 - ABFFF
V3	B0000 - B3FFF
Vx	BC000 - BFFFF

WAIT STATE LOGIC

Z8001 transactions can be lengthened to accommodate a slow device by the insertion of wait cycles. If an I/O operation or memory operation needs more time to complete a transaction, wait cycles are inserted between T2 and T3 by forcing the -WAIT input active (low). The Z8001 then waits for periods of 250ns.

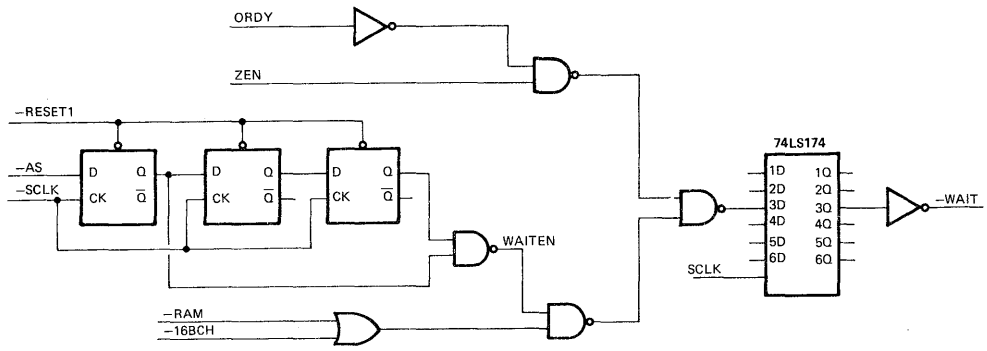


Fig. 8-8 Wait State Logic

The -WAIT signal goes low when

a) -RAM is high signalling that the memory is not being accessed and WAITEN high generated by the address strobe signal -AS going low. This means that an I/O cycle is still in progress.

b) ZEN is high meaning that the memory is enabled and ORDY is low signalling that the APB Z8000 board is not yet ready and so the memory cycle being performed needs to be lengthened.

I/O DEVICES

The I/O port addresses for I/O devices present on the Z8000 board are generated by the 3 to 8 line decoder, 74LS138, which decodes the Z8001 address lines ZA5-ZA15. The following table gives the address for each I/O port available:

Address Hex	Port Function	Port Select Signal
081	Interrupt Control Port	-ZI008X
0A1	Write Port for Various Controls	-ZI00AX
0CX	8251A Serial Interface	-ZI00CX
12X	8253 Timer	-ZI012X

The following is a description of the functions of these I/O ports. These ports are all attached to the Z8001 local data bus and are not accessible from the 8086 system processor.

Interrupt Control Port: Address '081'

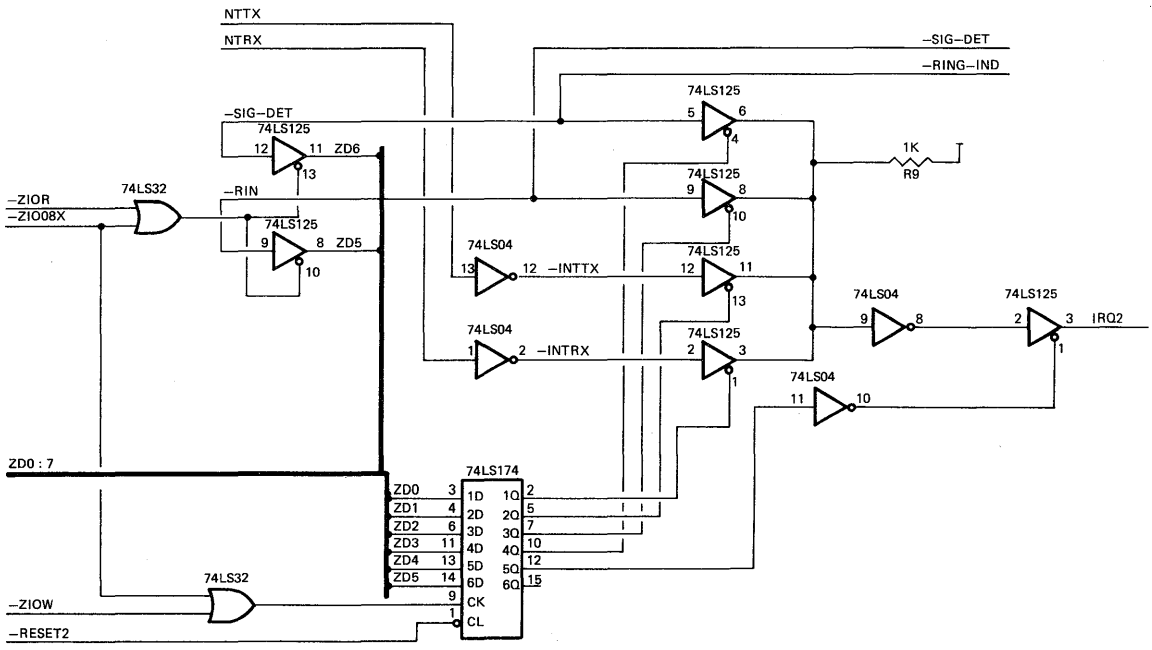


Fig. 8-9 Interrupt Control Port

The 8086 processor uses IRQ2 as the interrupt channel reserved for external boards like the Z8001 board. The Z8000 board contains four interrupt sources namely -SIG DET interrupt, -RING DET interrupt, 8251A TXRDY

interrupt and 8251A RXRDY interrupt. These four interrupts are all routed to the IRQ2 interrupt and with one or more of these interrupts active, IRQ2 goes active signalling an interrupt to the system.

During an I/O write cycle and an address of '081', the Z8001 data bits, ZD0-ZD4, enable/disable the four interrupts by driving the respective gates to a low or high impedance. The following table describes the function of each bit during an I/O write.

I/O Write	
ZD7-ZD5	Not Used
ZD4	0 = Disable IRQ2 1 = Enable IRQ2
ZD3	0 = Enable -SIG DET Interrupt 1 = Disable -SIG DET Interrupt
ZD2	0 = Enable -RING DET Interrupt 1 = Disable -RING DET Interrupt
ZD1	0 = Enable 8251A TXRDY Interrupt 1 = Disable 8251A TXRDY Interrupt
ZD0	0 = Enable 8251A RXRDY Interrupt 1 = Disable 8251A RXRDY Interrupt

On an I/O read operation, the signal -Z1008X low activates the tri-state drivers 74LS125 so that the two Z8001 data bits, ZD5 and ZD6, contain the state of the signals -RING IND and -SIG DET respectively as shown in the following table.

I/O Read	
ZD7	Not Used
ZD6	0 = -RING IND is active 1 = -RING IND is inactive
ZD5	0 = -SIG DET is active 1 = -SIG DET is inactive
ZD4-ZD0	Not Used

Note that all interrupts are enabled upon reset and the interrupts that need to be disabled are taken care of by the system software through this port.

Write Port for Various Controls: Address '0A1'

This port consists of the latch 74LS174 and during an I/O write cycle the outputs, -EN0 to -EN4 set the transmit/receive signals for the 8251A Serial Communication Interface according to the contents of the Z8001 data bus bits ZD0-ZD4 as shown in the following table. ZD5 generates the signal ZEN which enables or disables memory access.

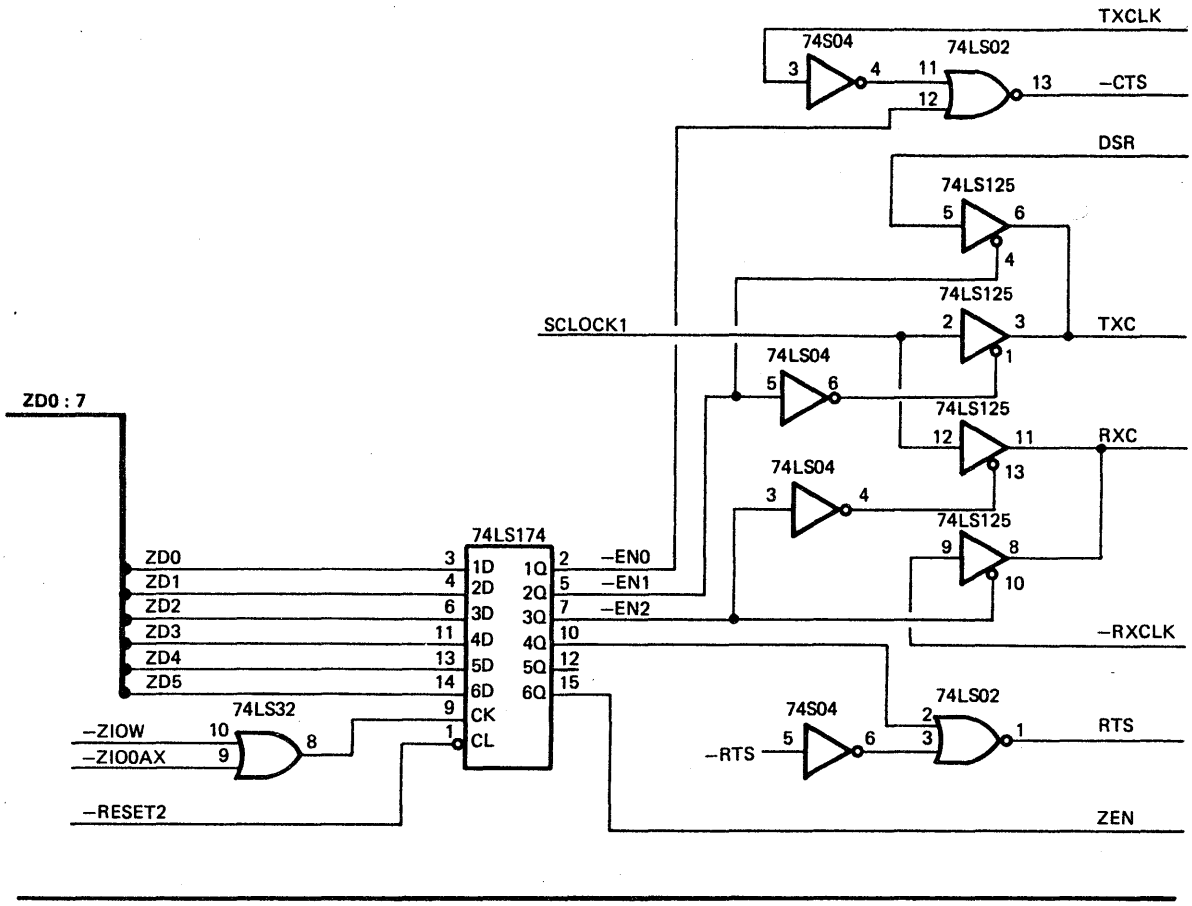


Fig. 8-10 I/O Control Port '0A1'

I/O Write		
ZD7-ZD6		Not Used
ZD5	ZEN	0 = Memory Access Disabled 1 = Memory Access Enabled
ZD4		Not Used
ZD3	-EN3	0 = RS232C RTS is from 8251A 1 = RS232C RTS is always enabled
ZD2	-EN2	0 = 8251A RXC is from external source 1 = 8251A RXC is from 8253 Timer
ZD1	-EN1	0 = 8251A TXC is from external source 1 = 8251A TXC is from 8253 Timer
ZD0	-EN0	0 = 8251A -CTS is from external source 1 = 8251A -CTS is always enabled

Note that on default ZD0-ZD7 are at a logic level 0.

8251A Control Port: Address '0CX'

The address '0CX' generates the signal -ZI00CX which is then used by the 8251A Serial Communication Controller as the chip select signal. The 8251A uses the Z8001 address bit ZA1 to determine whether it is the data register or the status register that communicates with the Z8001 data bus during a Z8001 I/O read/write cycle. ZA1 low selects the data register while ZA1 high selects the status register. The direction of data depends on the signals -ZIOR and -ZIOW.

I/O Write		
Address	ZA1	
0C1	0	Data Bus ——> 8251A Data Register
0C3	1	Data Bus ——> 8251A Status Register

I/O Read		
Address	ZA1	
0C1	0	8251A Data Register ——> Data Bus
0C3	1	8251A Status Register ——> Data Bus

8253 Timer Control Port: Address '012X'

The address '012X' generates the signal -ZI012X which is then used by the 8253 Timer as the chip select signal. The 8253 uses the Z8001 address bits, ZA1 and ZA2, to select the counter to be loaded or written from/to the data bus during a Z8001 I/O read/write cycle. The following table illustrates this.

I/O Write			
Address	ZA2	ZA1	
121	0	0	Load 8253 Counter No. 0
123	0	1	Load 8253 Counter No. 1
125	1	0	Load 8253 Counter No. 2
127	1	1	Load 8253 Mode Word

I/O Read			
Address	ZA2	ZA1	
123	0	1	Read 8253 Counter No. 1
125	1	0	Read 8253 Counter No. 2
127	1	1	No Operation

The clock used by the timer is a 1.23 MHz clock derived from the 19.66 MHz Z8001 clock. The three counters of this timer are used as follows:

- Counter 0 output is normally used as the receive and transmit clock of the 8251 communication controller.

- Counter 1 output is currently not used.

- Counter 2 output is used as real time clock and is latched into a flipflop to cause a Non-Vector Interrupt, -NVI, to the Z8001 processor.

Z8001 Port: Address '80C1

This port, unlike the previously mentioned ports, is accessible by both the 8086 and Z8001 processors. The 3 to 8 line decoder 74LS138 decodes address bits A4-A9 and A15 and when an address of '80C1' is present the signal -IOCX goes active low. This enables the Z8001 port which consists mainly of the latch 74LS175. During an I/O write cycle, the latch is clocked and thus the data bits D0, D6 and D7 are latched to the outputs. D0 causes the Z8001 board circuitry to be reset when low and activates the circuitry when high. D6 low sets the circuitry in an M20 compatible mode while D6 high sets the circuitry in an M30 compatible mode which is not currently used. The other data bits are not used.

I/O Write	
D7	Not Used
D6	0 = M20 Compatible Mode 1 = M30 Compatible Mode
D5-D1	Not Used
D0	0 = Reset Z8001 Board 1 = Activate Z8001 Board

During an I/O read cycle, -IOR low and -IOCX low, activate the driver 74LS125 and thus D6 depends on the setting of the jumper W1. D6 is set low for a 4 colour system and high for an 8 colour system.

I/O Read	
D7	Not Used
D6	0 = 4 colour system 1 = 8 colour system
D5-D0	Not Used

Other I/O Devices:

The Z8000 board uses other I/O devices present on the system motherboard. The addresses for these I/O devices are the same as the ones for the system I/O devices plus '8000'. The reason for this is that the Z8001 must be able to distinguish between on board and off board I/O devices. The following is a list of the I/O devices present on the system and their addresses using the Z8000 board.

Address Range	Device/Function
8000-800F	8237A-4 DMA Controller
8020-8021	8259A Interrupt Controller
8040-8043	8253-5 Timer
8050-8053	8530 SCC
8060	8041 Keyboard Data Read Only
8061	Control Port A
8062	Input Port B
8064	8041 Command, Write Only
8065	Communication Input Port
8066	System Configuration,
8067	Read Only
8080-8083	DMA Segment Register
80C1	Z8000 Board
80F0-808E	System ID PROM
8320-8323	Hard Disk Controller
8378-837B	Parallel Printer Port
83F2	Floppy Disk Controller Port
83F4-83F5	Floppy Disk Controller
83F8-83FF	Primary Asynchronous Controller

8251A SERIAL INTERFACE

The Z8000 board contains a serial communication interface which is used together with the one present on the motherboard. It is based on the 8251 Programmable Communication Interface and provides an RS232 interface connector.

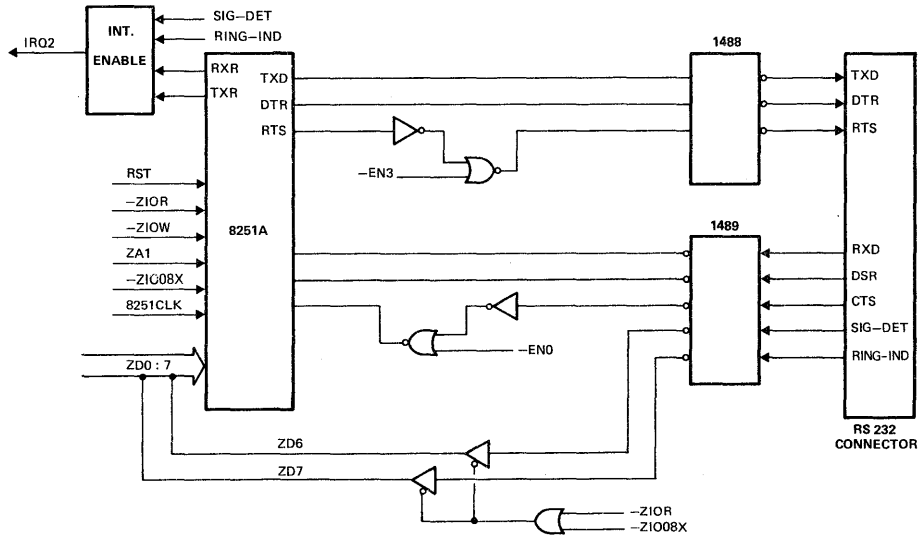


Fig. 8-11 Serial Communication Interface

The 8251A is a Universal synchronous/asynchronous receiver/transmitter (USART). It performs serial to parallel conversion of data on the received data circuit, parallel to serial conversion on the transmitted data circuit and the modem control functions. It also has a status reporting capability as well as error detecting. Essentially the USART has the same features as the 8250 Asynchronous communication element used on the motherboard and described in chapter 2. But it also incorporates additional features like double buffered data paths with separate I/O registers, automatic break detection and handling, and false start bit detection. The 8251A status can be read at any time but the status update is inhibited during status read.

The 8251 uses a 2 MHz clock generated from the 8 MHz clock, CLK8, coming from the system motherboard.

The I/O ports described earlier in the I/O devices section control the function of the 8251 Programmable Communication Interface. The PCI communication with the Z8001 processor is done through the Z8001, 8 bit, data bus, ZD0-ZD7.

The 8251A PCI has a set of transmit/receive control inputs and outputs that are used to produce an RS 232 serial interface. These signals pass through the line drivers 1488 and 1489 before/after they go to the RS 232 connector.

Similar to the M20 PCOS, the M24 PCOS can manage three serial communication ports:

COM 0 - RS232 connector on the APB Z8000 Board

COM 1 - RS232 connector on the motherboard

COM 2 - RS232 connector on the optional serial communications board

SERIAL INTERFACE CONNECTOR

The EIA RS-232-C interchange circuit signals are made available through a 25 pin , D shell connector. This connector is on the edge of the APB Z8000 board and protrudes through one of the expansion slots on the rear panel of the system module.

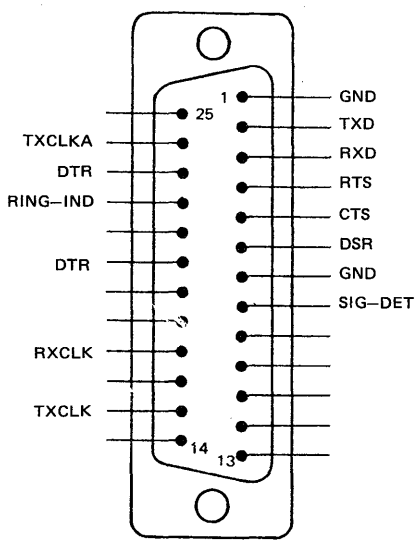


Fig. 8-12 Serial Interface Connector

RS 232 Connector Signals

The following is a description of the signals available on the RS 232 interface connector. I/O direction is with respect to the motherboard.

Interchange Signal	Cct	I/O	Description
PRGND	AA	-	Protective Ground - Connected to Basic Module frame.
GND	AB	-	Signal Ground/Common Return - Common ground reference for interchange circuits, except Protective Ground.
TxD	BA	O	Transmitted Data, to DCE - Generated by data terminal equipment and transferred to local modem or data set for transmission over the communication channel to the remote data terminal equipment.
RxD	BB	I	Received Data, from DCE - Generated by local modem or data set in response to data signals received over the communication channel from remote data terminal equipment .
RTS	CA	O	Request to Send, to DCE - Used to condition the local modem or data set for data transmission.
CTS	CB	I	Clear to Send, from DCE - Used to indicate whether or not the modem or data set is ready to transmit data.
DSR	CC	I	Data Set Ready, from DCE - Used to indicate the status of the local modem or data set.
DTR	CD	O	Data Terminal Ready, to DCE - Used to control the switching of modem or data set to the communication channel.
RING_IND	CD	I	Ring Indicator, from DCE - Indicates that a ringing signal is being received on the communication channel.
SIG_DET	CF	I	Receive Line Signal Detect (Data Carrier Detected), from DCE - Used to indicate that the data carrier has been detected by the modem or data set.

A. DISPLAY CONTROLLER PAL AND ROM DESCRIPTIONS

CONTENTS

PAGE

This appendix gives a detailed description of the Programmable Array Logic circuits as well as the contents of the EPROMS used in the Display Controller.

PAL 10L8 DESCRIPTION

The following is a detailed description of the PAL 10L8 chip for rev. P2 boards.

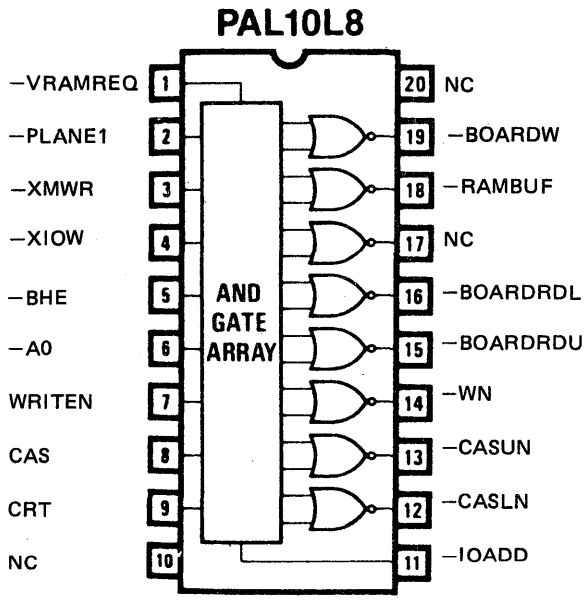
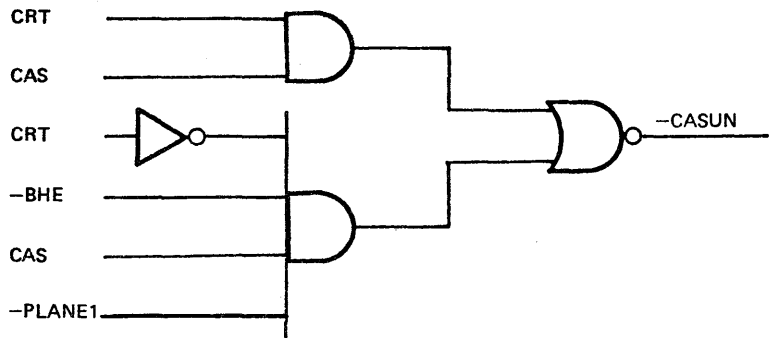
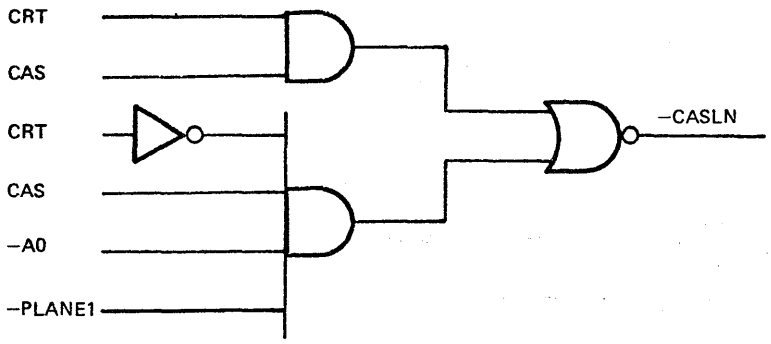


Fig. A-1 PAL 10L8 Pin Functions for Rev. P2 Boards

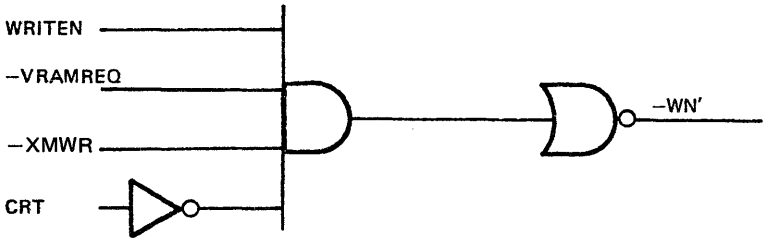
–CASUN: ACTIVE LOW SIGNAL WHICH ACTS AS THE COLUMN ADDRESS STROBE FOR THE UPPER BYTE



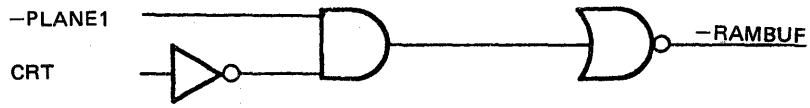
–CASLN: ACTIVE LOW SIGNAL WHICH ACTS AS THE COLUMN ADDRESS STROBE FOR THE LOWER BYTE



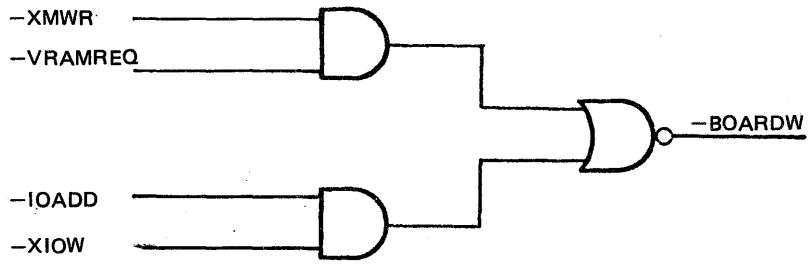
–WN: ACTIVE LOW SIGNAL WHICH ENABLES WRITE OPERATIONS TO BE PERFORMED INTO THE DISPLAY MEMORY



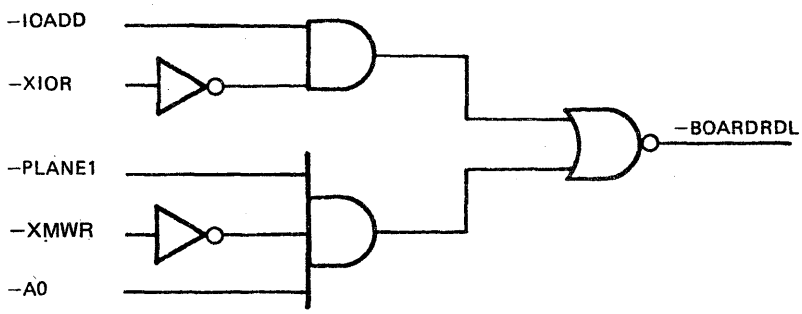
−RAMBUF: ACTIVE LOW SIGNAL WHICH ENABLES THE CPU DATA BUFFER



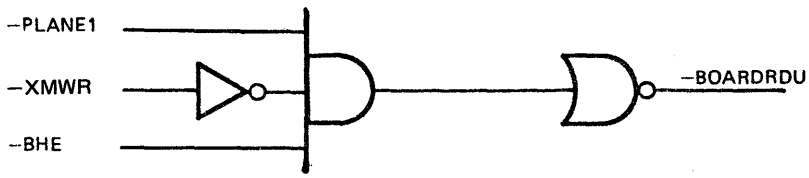
−BOARDW: ACTIVE LOW SIGNAL WHICH IS ACTIVE DURING WRITE OPERATIONS



−BOARDRDL: ACTIVE LOW SIGNAL WHICH ENABLES THE READ/WRITE DATA LATCH FOR THE LOWER BYTE



−BOARDRDU: ACTIVE LOW SIGNAL WHICH ENABLES THE READ/WRITE DATA LATCH FOR THE UPPER BYTE



The following is a detailed description of the PAL 10L8 chip for rev. P3 boards.

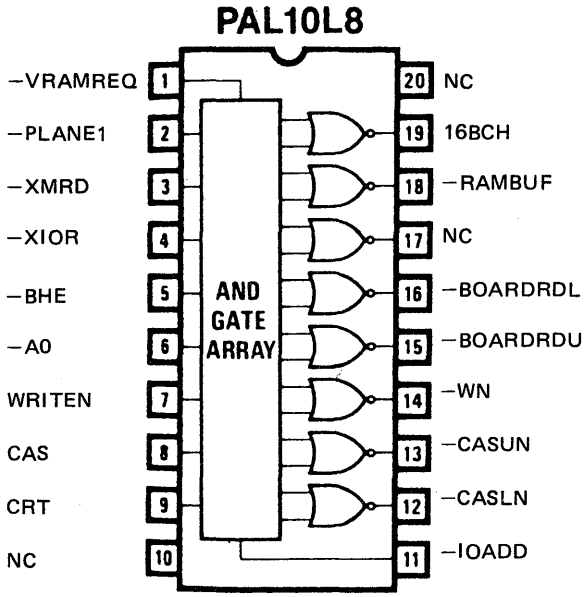
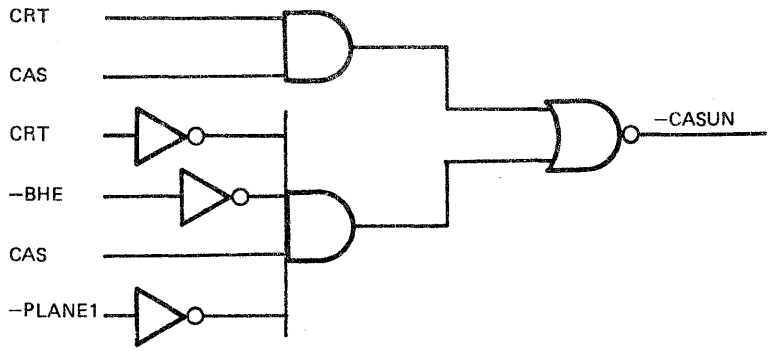
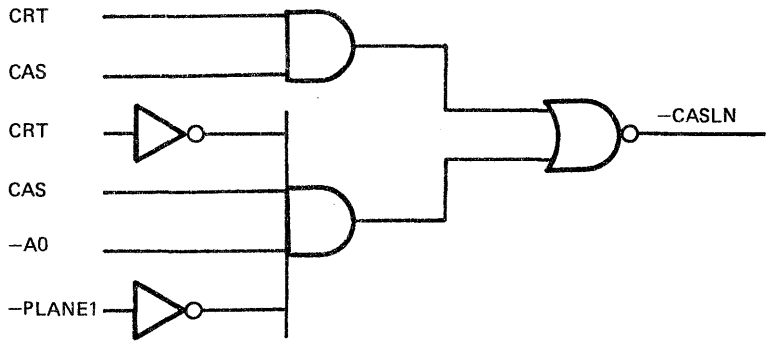


Fig. A-2 PAL 10L8 Pin Functions

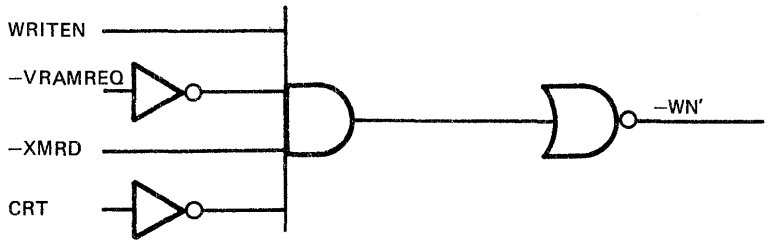
–CASUN: ACTIVE LOW SIGNAL WHICH ACTS AS THE COLUMN ADDRESS STROBE FOR THE UPPER BYTE



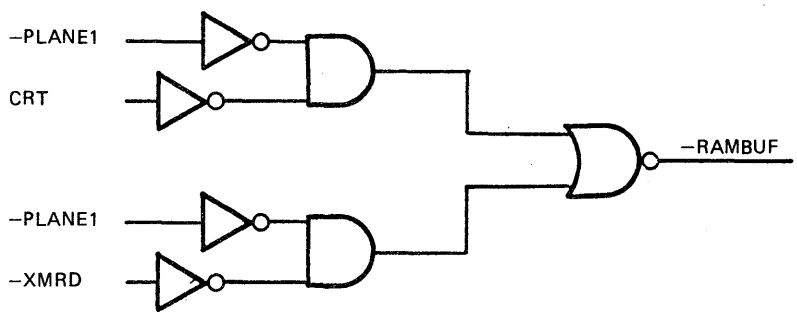
–CASLN: ACTIVE LOW SIGNAL WHICH ACTS AS THE COLUMN ADDRESS STROBE FOR THE LOWER BYTE



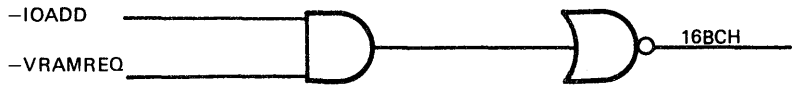
–WN: ACTIVE LOW SIGNAL WHICH ENABLES WRITE OPERATIONS TO BE PERFORMED INTO THE DISPLAY MEMORY



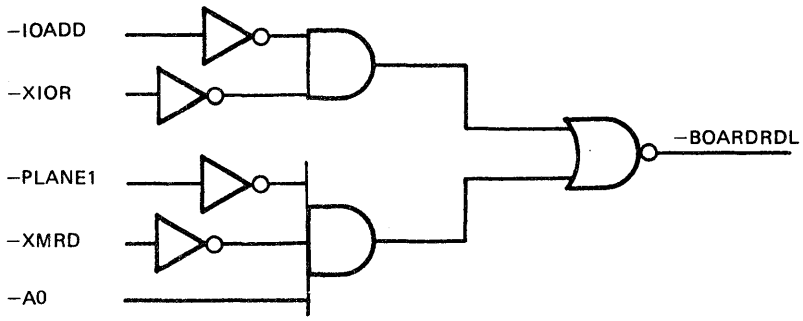
–RAMBUF: ACTIVE LOW SIGNAL WHICH ENABLES THE CPU DATA BUFFER



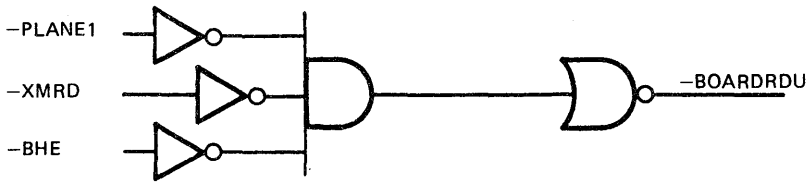
16BCH: ACTIVE HIGH SIGNAL WHICH INDICATES TO THE CPU THAT A 16 BIT CHANNEL IS BEING USED



–BOARDRDL: ACTIVE LOW SIGNAL WHICH ENABLES THE READ/WRITE DATA LATCH FOR THE LOWER BYTE



–BOARDRDU: ACTIVE LOW SIGNAL WHICH ENABLES THE READ/WRITE DATA LATCH FOR THE UPPER BYTE



PROM 27S19A CONTENTS

The following table shows the contents of PROM 27S19A for rev. P2 boards

	Q7	Q6	Q5	Q4	Q3	Q2	Q1	Q0	
ADDRESS	MUXSWCH	WRITEN	-MUXSWCH	-6845CLK	6845CLK	-SHFLD	SHFLD	-RAS	HEX DATA
00	0	0	1	0	1	1	0	1	2D
01	0	0	1	0	1	1	0	0	2C
02	1	0	0	1	0	1	0	0	94
03	1	0	0	1	0	1	0	0	94
04	1	0	0	1	0	0	1	0	92
05	0	0	1	1	0	0	1	0	32
06	0	0	1	0	1	1	0	1	2D
07	0	0	1	0	1	1	0	1	2D
08	0	0	1	0	1	1	0	1	2D
09	0	1	1	0	1	1	0	0	6C
0A	1	1	0	0	1	1	0	0	CC
0B	1	1	0	0	1	1	0	0	CC
0C	1	1	0	0	1	1	0	0	CC
0D	0	1	1	0	1	1	0	0	6C
0E	0	1	1	0	1	1	0	1	6D
0F	0	0	1	0	1	1	0	1	2D
10	0	0	1	0	1	1	0	1	2D
11	0	0	1	0	1	1	0	0	2C
12	1	0	0	1	0	1	0	0	94
13	1	0	0	1	0	1	0	0	94
14	1	0	0	1	0	1	0	0	94
15	0	0	1	1	0	0	1	0	32
16	0	0	1	0	1	1	0	1	2D
17	0	0	1	0	1	1	0	1	2D
18	0	0	1	0	1	1	0	1	2D
19	0	1	1	0	1	1	0	0	6C
1A	1	1	0	1	0	1	0	0	D4
1B	1	1	0	1	0	1	0	0	D4
1C	1	1	0	1	0	1	0	0	D4
1D	0	1	1	1	0	0	1	0	72
1E	0	1	1	0	1	1	0	1	6D
1F	0	0	1	0	1	1	0	1	2D

The following table shows the contents of PROM 27S19A for rev. P3 boards.

	Q7	Q6	Q5	Q4	Q3	Q2	Q1	Q0	
ADDRESS	MUXSWCH	WRITEN	-MUXSWCH	-6845CLK	6845CLK	ARBCLK	SHFLD	-RAS	HEX DATA
00	0	0	1	0	1	1	0	1	2D
01	0	0	1	0	1	1	0	0	2C
02	1	0	0	1	0	1	0	0	94
03	1	0	0	1	0	1	0	0	94
04	1	0	0	1	0	0	0	1	92
05	0	0	1	1	0	1	1	0	36
06	0	0	1	0	1	1	0	1	2D
07	0	0	1	0	1	1	0	1	2D
08	0	0	1	0	1	1	0	1	2D
09	0	1	1	0	1	1	0	0	6C
0A	1	1	0	0	1	1	0	0	CC
0B	1	1	0	0	1	1	0	0	CC
0C	1	1	0	0	1	1	0	0	CC
0D	0	1	1	0	1	1	0	0	6C
0E	0	1	1	0	1	1	0	1	6D
0F	0	0	1	0	1	1	0	1	2D
10	0	0	1	0	1	1	0	1	2D
11	0	0	1	0	1	1	0	0	2C
12	1	0	0	1	0	1	0	0	94
13	1	0	0	1	0	1	0	0	94
14	1	0	0	1	0	0	0	0	90
15	0	0	1	1	0	1	1	0	36
16	0	0	1	0	1	1	0	1	2D
17	0	0	1	0	1	1	0	1	2D
18	0	0	1	0	1	1	0	1	2D
19	0	1	1	0	1	1	0	0	6C
1A	1	1	0	1	0	1	0	0	D4
1B	1	1	0	1	0	1	0	0	D4
1C	1	1	0	1	0	1	0	0	D4
1D	0	1	1	1	0	1	1	0	76
1E	0	1	1	0	1	1	0	1	6D
1F	0	0	1	0	1	1	0	1	2D

SCRAMBLER EPROM

The following table gives the contents of the Scrambler EPROM for rev. P2 boards:

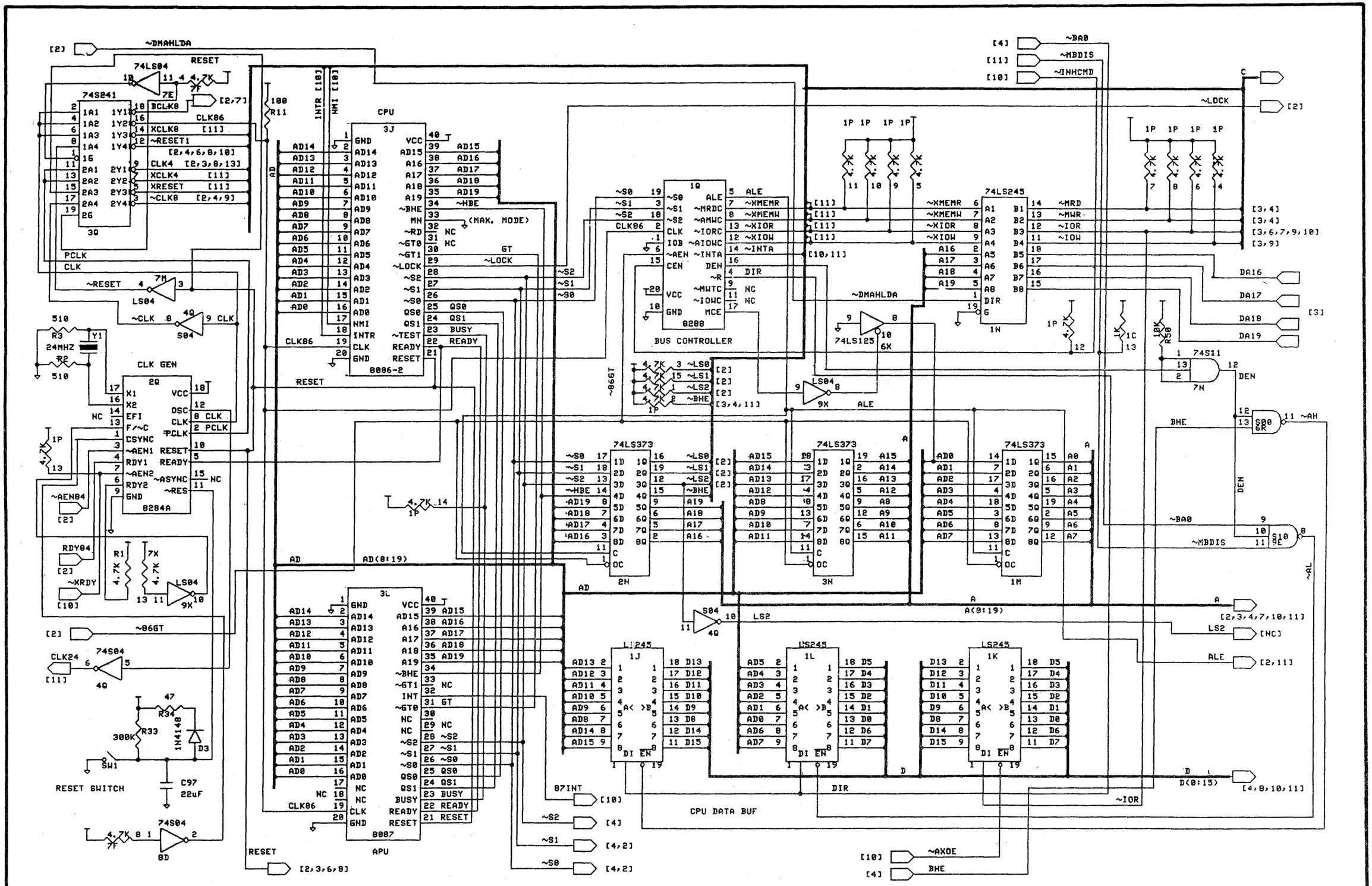
ADDRESS	VALUE	FUNCTION
0 - 55	39	R0 -80x25
56 - 7F	73	R0 80x25
80 - FF	0-7F	R0 TRANSPARENT
100 - 17F	0-7F	R1
180 - 1FF	0-7F	R1
200 - 27F	2-81	R2
280 - 2FF	0-7F	R2
300 - 309	06	R3
30A	09	R3
30B	06	R3
30C - 30F	0C	R3
310 - 3FE	09	R3
3FF	06	R3
400 - 440	1A	R4 AlPHA
441 - 4FF	6B	R4 GRAPHICS
500 - 5FF	0-FF	R5
600 - 6FF	0-FF	R6
700 - 740	19	R7 ALPHA
741 - 7FF	64	R7 GRAPHICS
800 - 802	00	R8
803	03	R8 (INTERLACED)
804 - 8FF	00	R8
900 - 902	1-5	R9 (ADDR*2)+1)
903	03	R9
904 - 97F	9-FF	R9 ((ADDR*2)+1)
980 - 9FF	1-FF	R9 ((ADDR*80)*2)+1)
A00 - A7F	*	R10 (* IF ADDRESS MOD 20<10 THEN VALUE = (ADDRESS-A00)+ ADDRESS MOD 20 ELSE VALUE = (ADDRESS-A00)-ADDRESS MOD 20)+10)
A80 - AFF	**	R10 (* IF ADDRESS MOD 20<8 THEN VALUE = (ADDRESS-A80)+ ADDRESS MOD 20 ELSE VALUE = ((ADDRESS-A80)-ADDRESS MOD 20)+10)
B00 - B7F	0-FE	R11 ((ADDR-B00)*2)
B80 - BFF	0-FE	R11 ((ADDR-B80)*2)
C00 - FFF	***	R12-15 (*** VALUE=ADDRESS MOD 100)

The following table gives the contents of the Scrambler EPROM (rev. SCRAM8) for rev. P3 boards:

ADDRESS	VALUE	FUNCTION
0 - 55	39	R0 -80x25
56 - 7F	73	R0 80x25
80 - FF	0-7F	R0 TRANSPARENT
100 - 1FF	0-FF	R1
200 - 27F	2-81	R2
280 - 2FF	0-7F	R2
300 - 3FF	0C	R3
400 - 440	1A	R4 AlPHA
441 - 47F	6B	R4 GRAPHICS
480 - 4FF	0-7F	R4 TRANSPARENT
500 - 57F	0	R5
580 - 5FF	0-7F	R5 TRANSPARENT
600 - 6FF	0-FF	R6
700 - 740	19	R7 ALPHA
741 - 77F	64	R7 GRAPHICS
780 - 7FF	0-7F	R7 TRANSPARENT
800 - 802	00	R8
803	03	R8 INTERLACED
804 - 87F	00	R8
880 - 8FF	0-7F	R8 TRANSPARENT
900 - 97F	1-FF	R9 (ADDR*2)+1
980 - 9FF	0-7F	R9 TRANSPARENT
A00 - A7F	*	R10 (* IF ADDRESS MOD 20<10 THEN VALUE = (ADDRESS-A00)+ ADDRESS MOD 20 ELSE VALUE = (ADDRESS-ADDRESS MOD 20)+10
A80 - AFF	0-7F	R10 TRANSPARENT
B00 - B7F	0-FE	R11 (STEP 2)
B80 - BFF	0-7F	R11 TRANSPARENT
C00 - FFF	**	R12-15 (** VALUE=ADDRESS MOD 100)

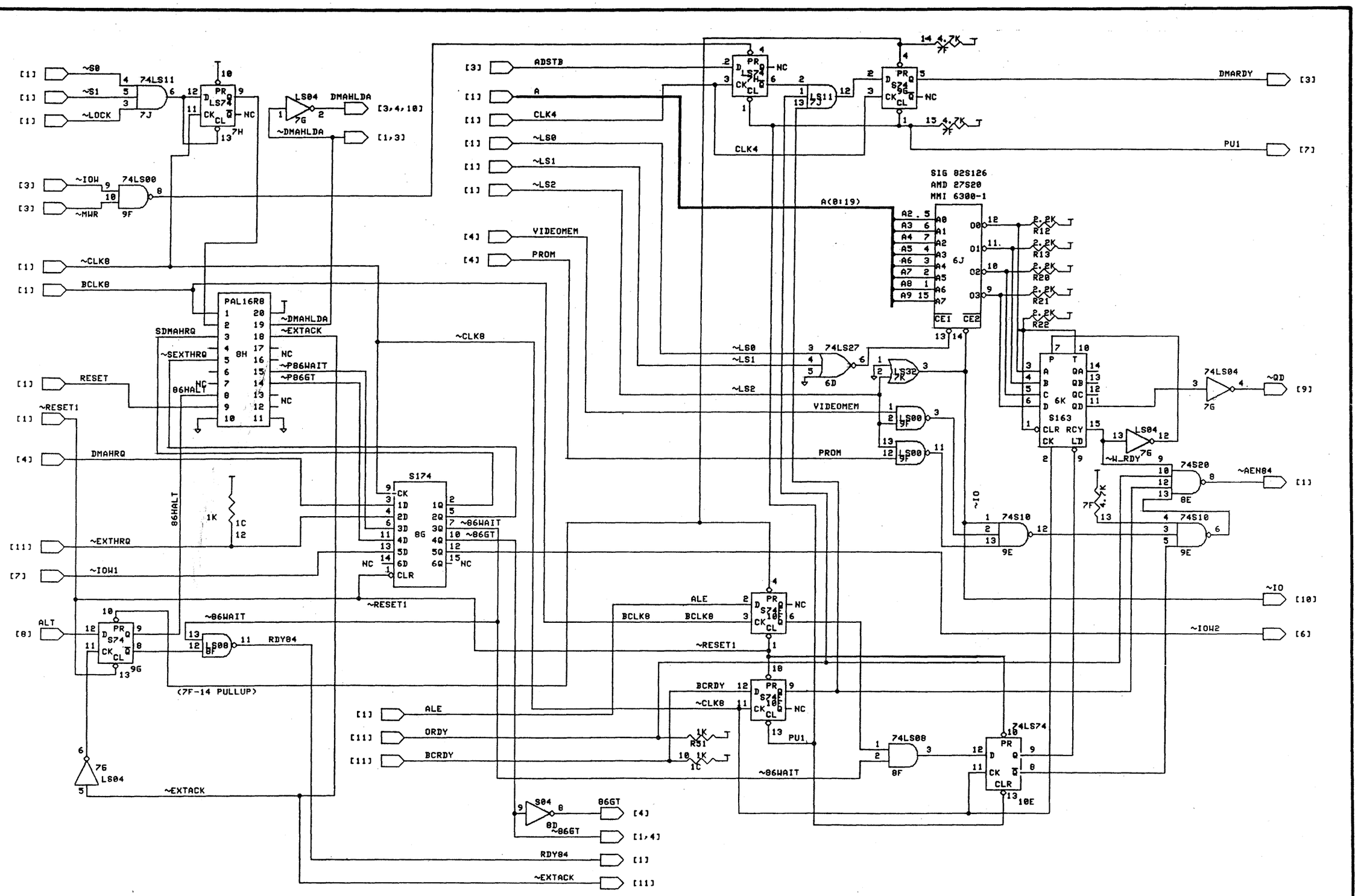
B. LOGIC DIAGRAMS

CONTENTS



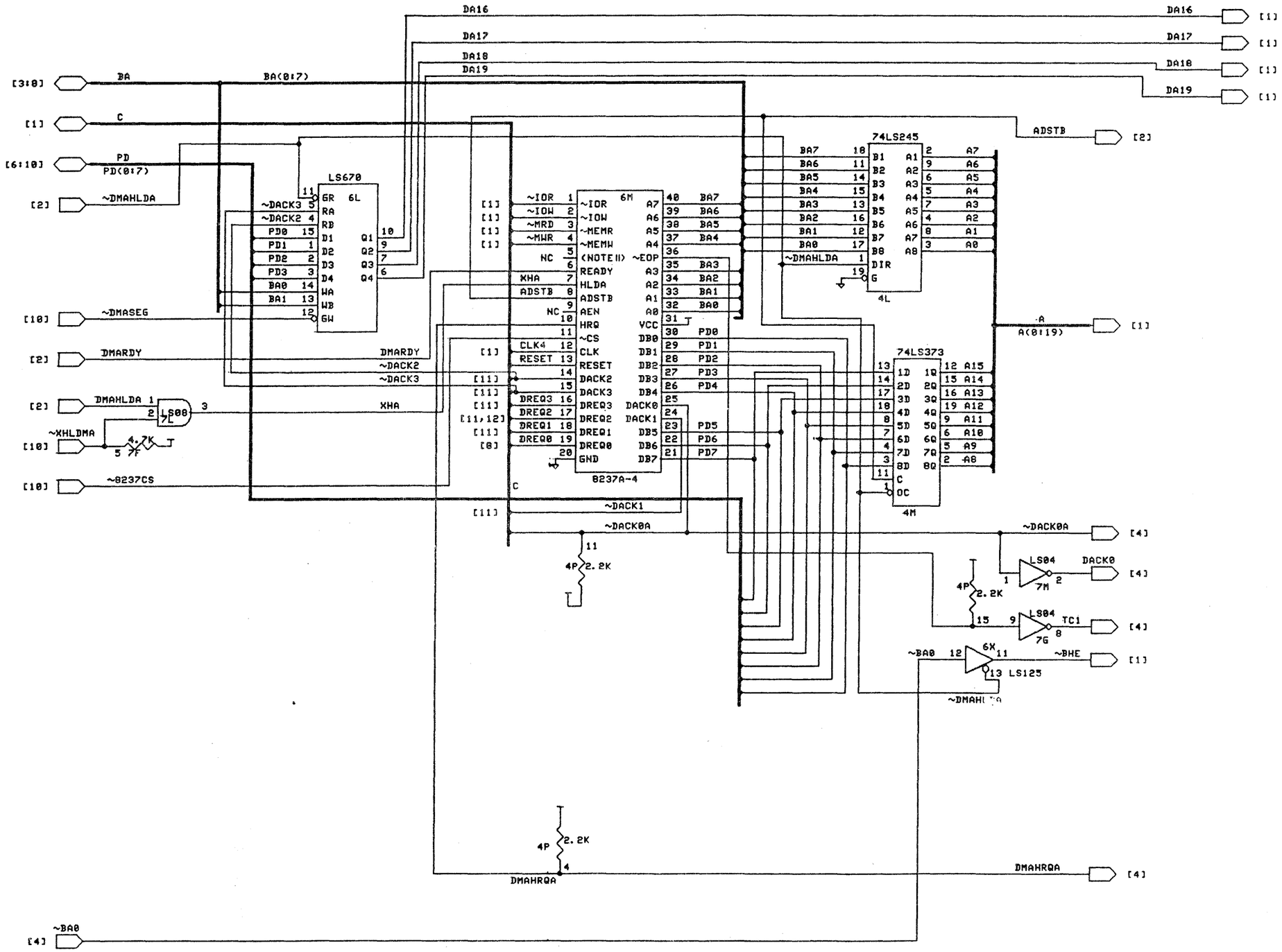
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
MOTHERBOARD
REV. P5 PAGE 1 OF 13



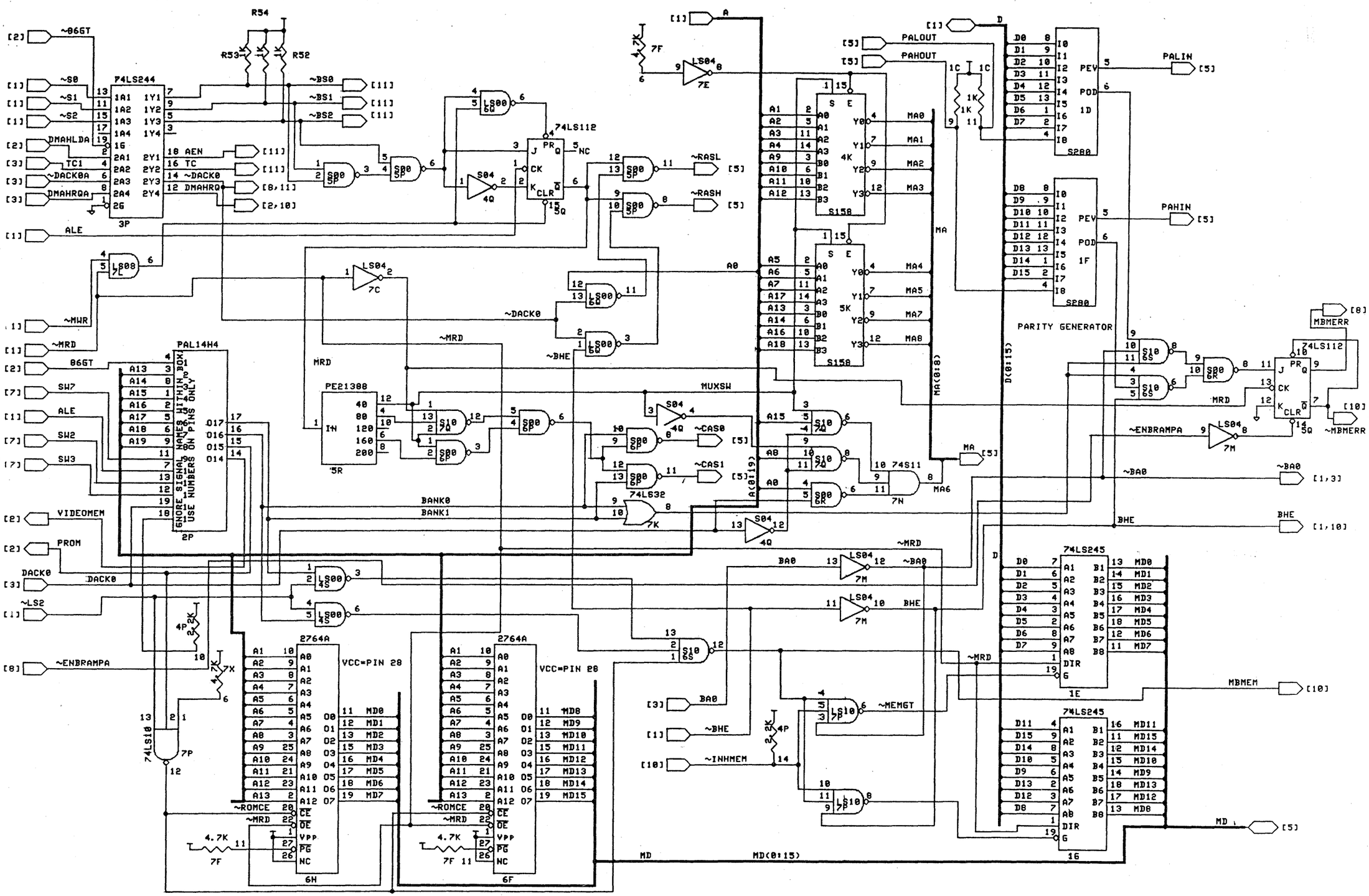
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
MOTHERBOARD
REV. P5 PAGE 2 OF 13



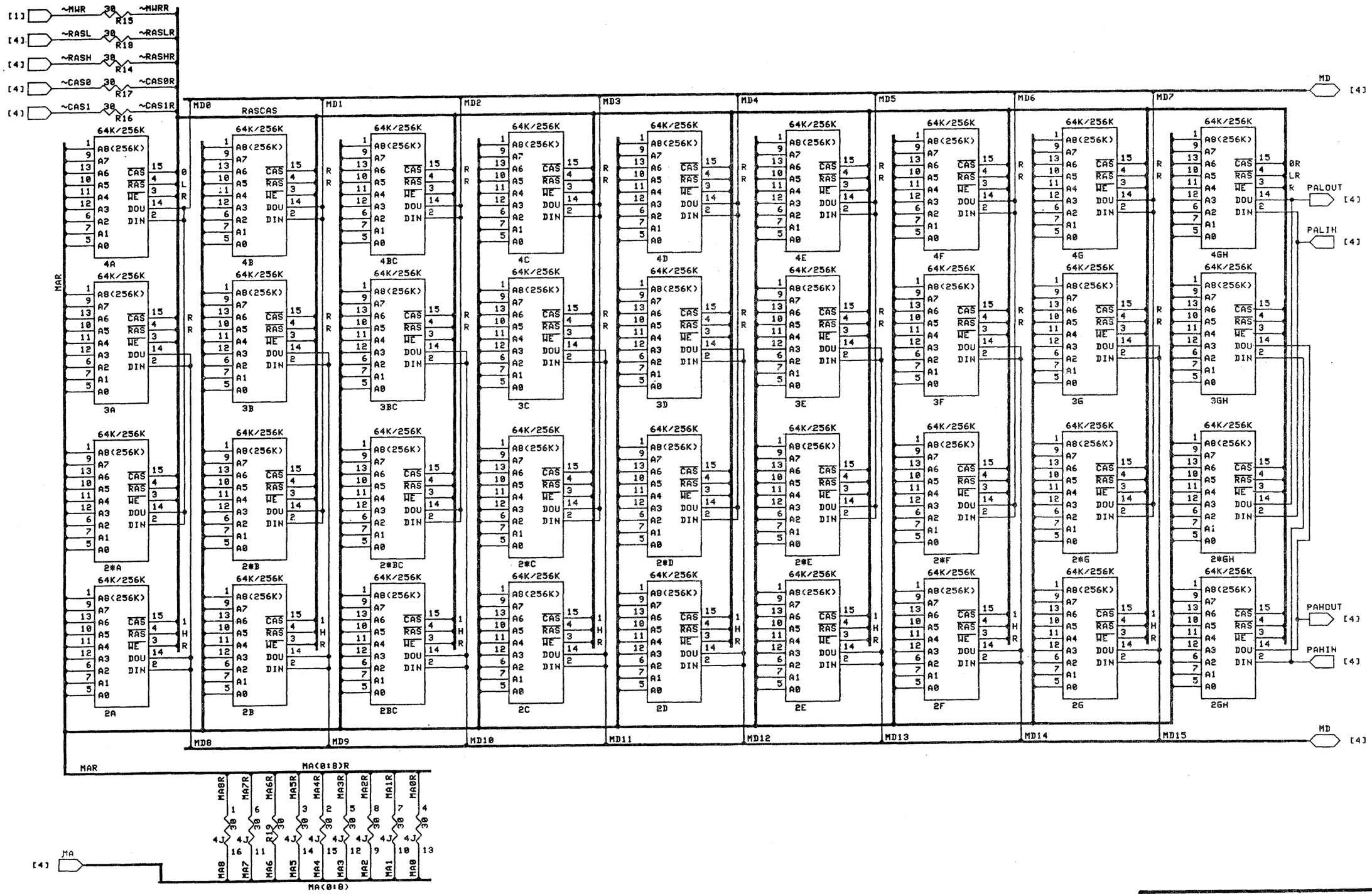
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
MOTHERBOARD
REV. P5 PAGE 3 OF 13



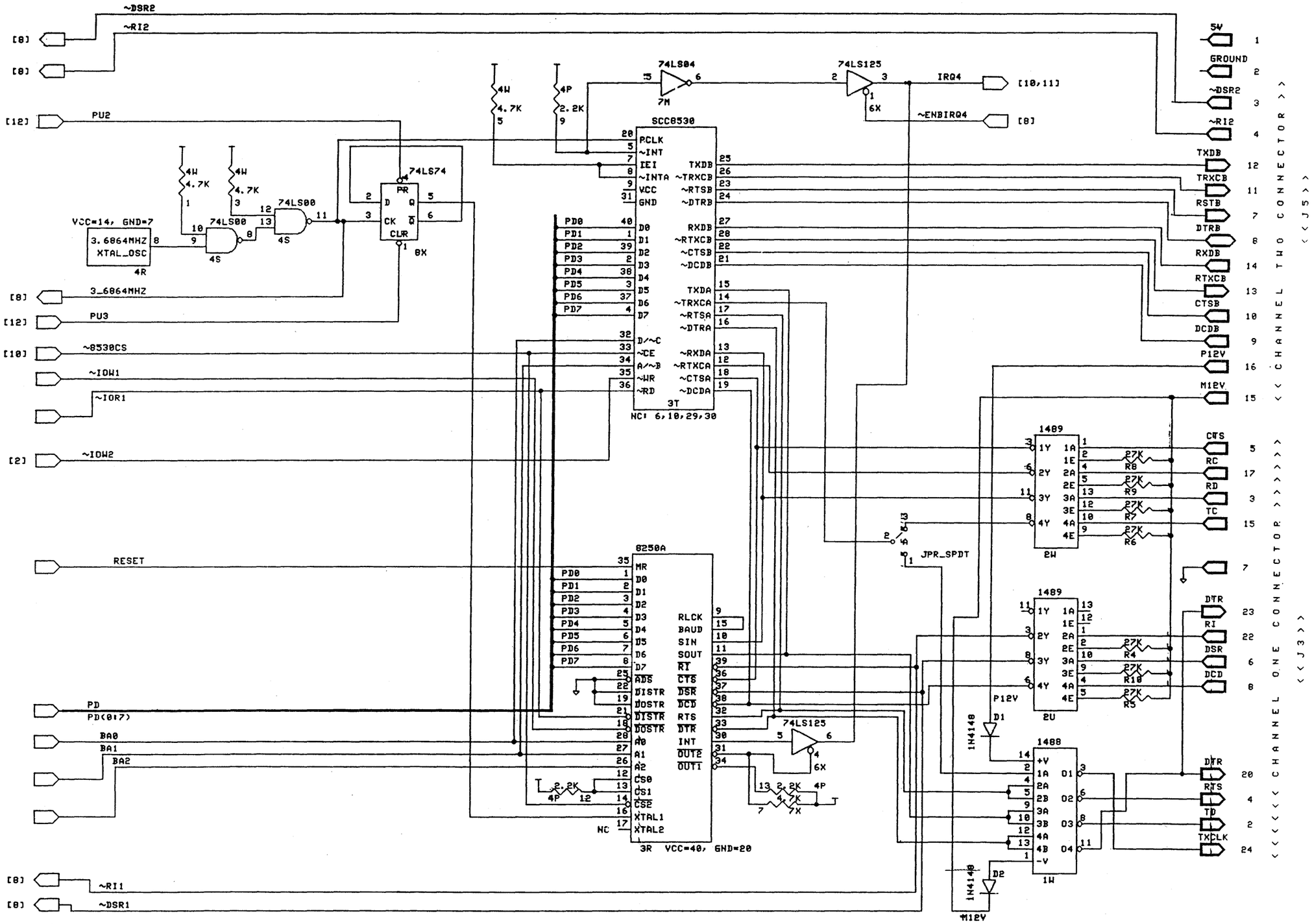
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
MOTHERBOARD
REV. P5 PAGE 4 OF 13



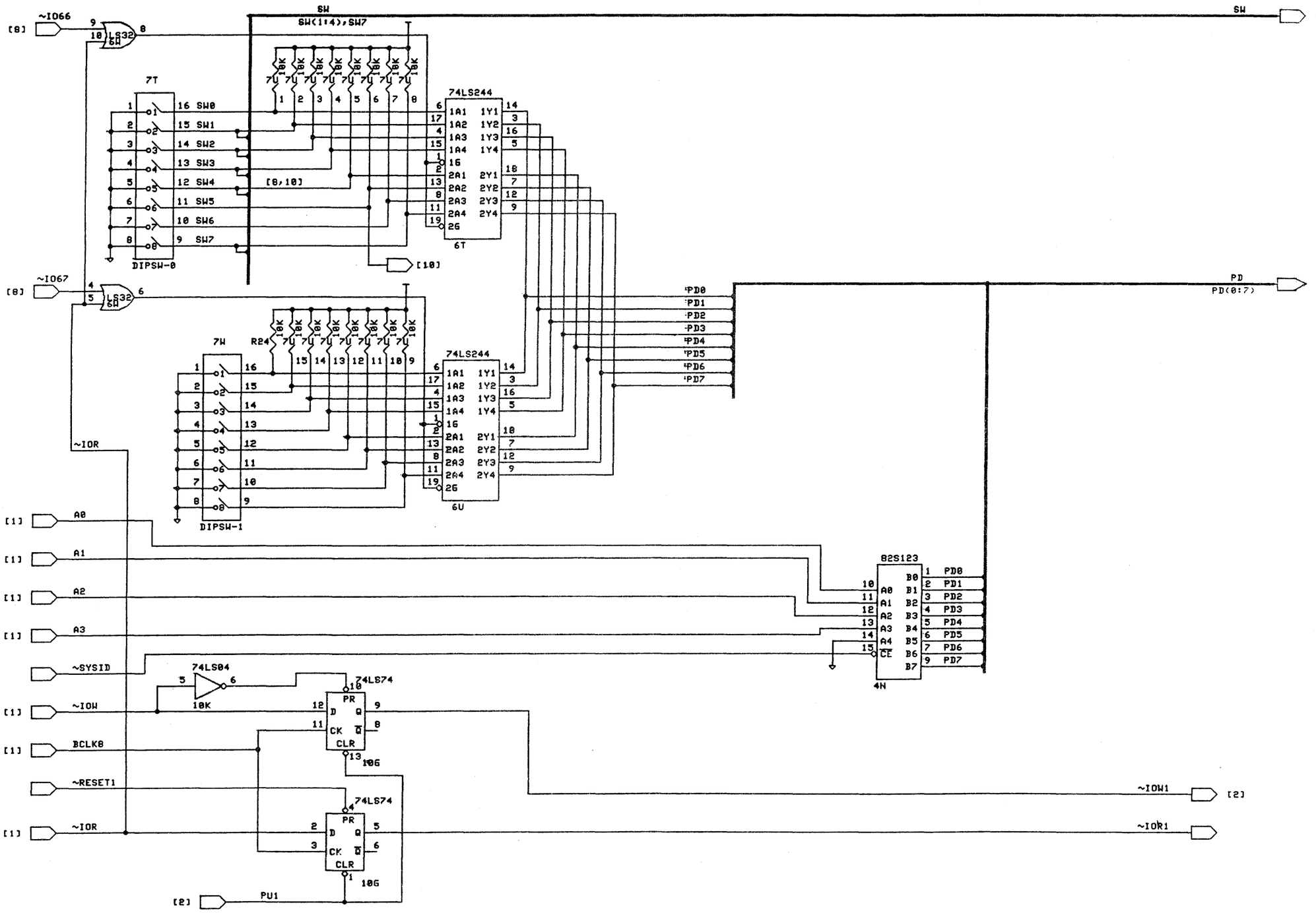
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
MOTHERBOARD
REV. P5 PAGE 5 OF 13



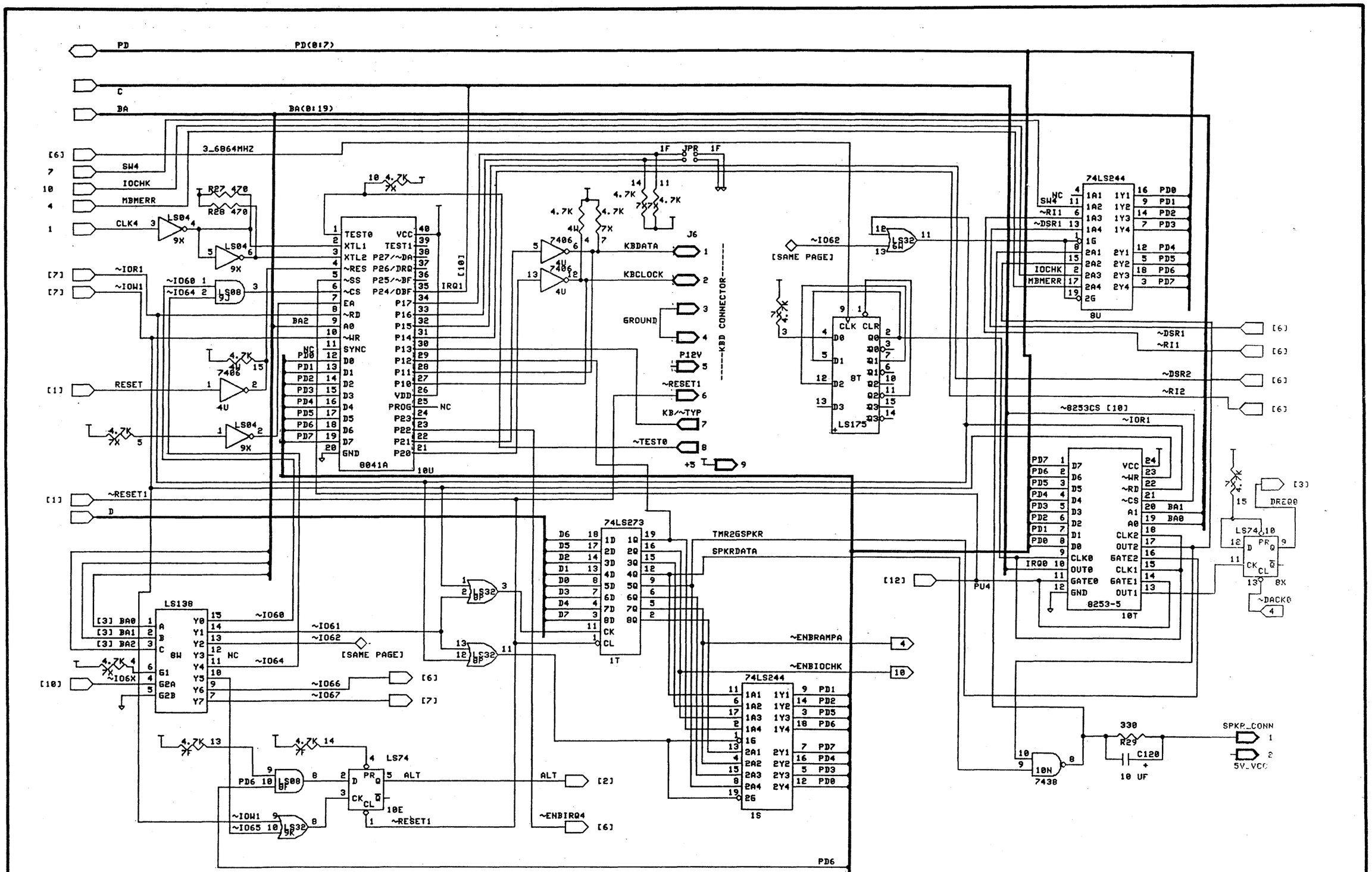
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
MOTHERBOARD
REV. P5 PAGE 6 OF 13



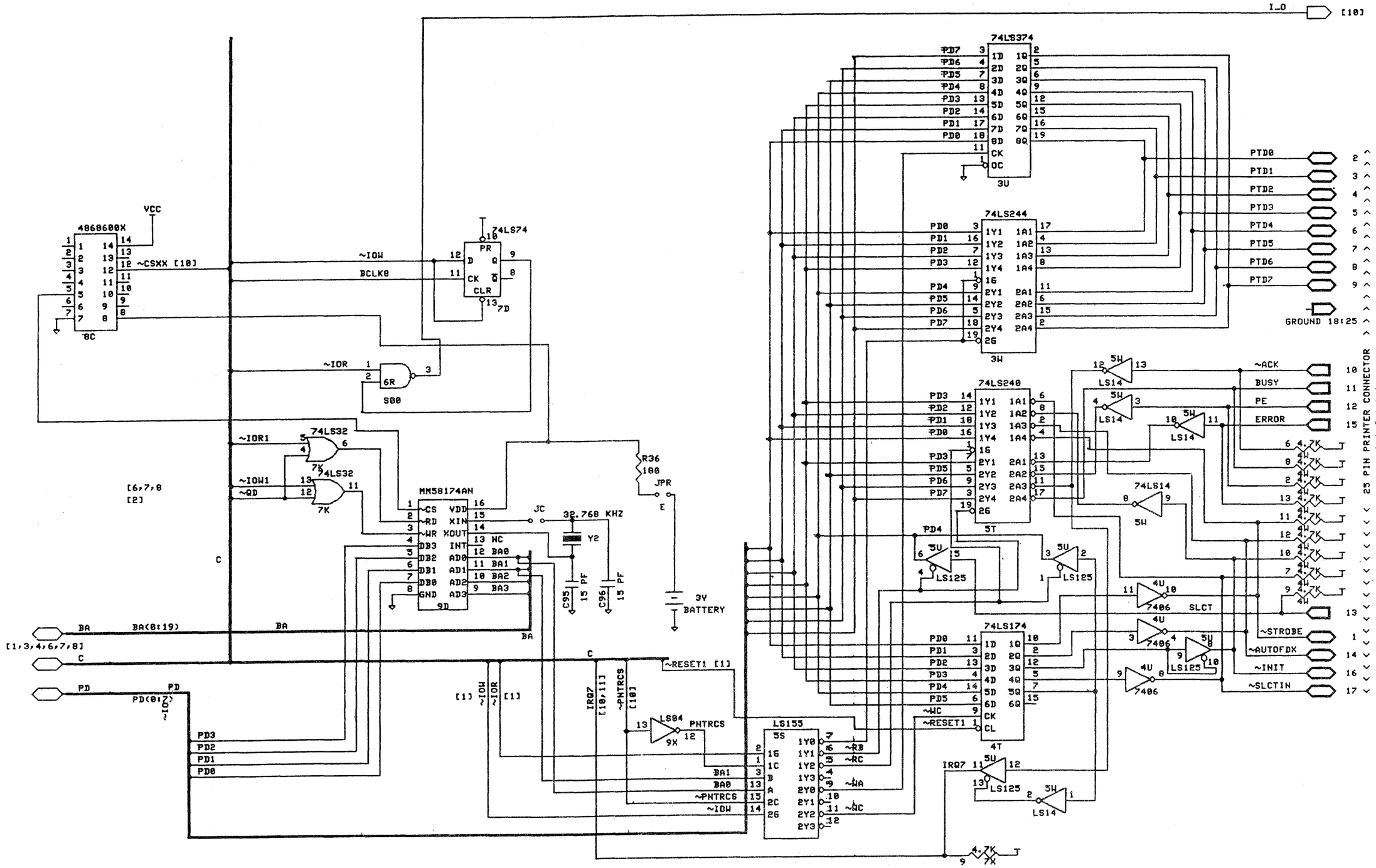
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
MOTHERBOARD
REV. P5 PAGE 7 OF 13



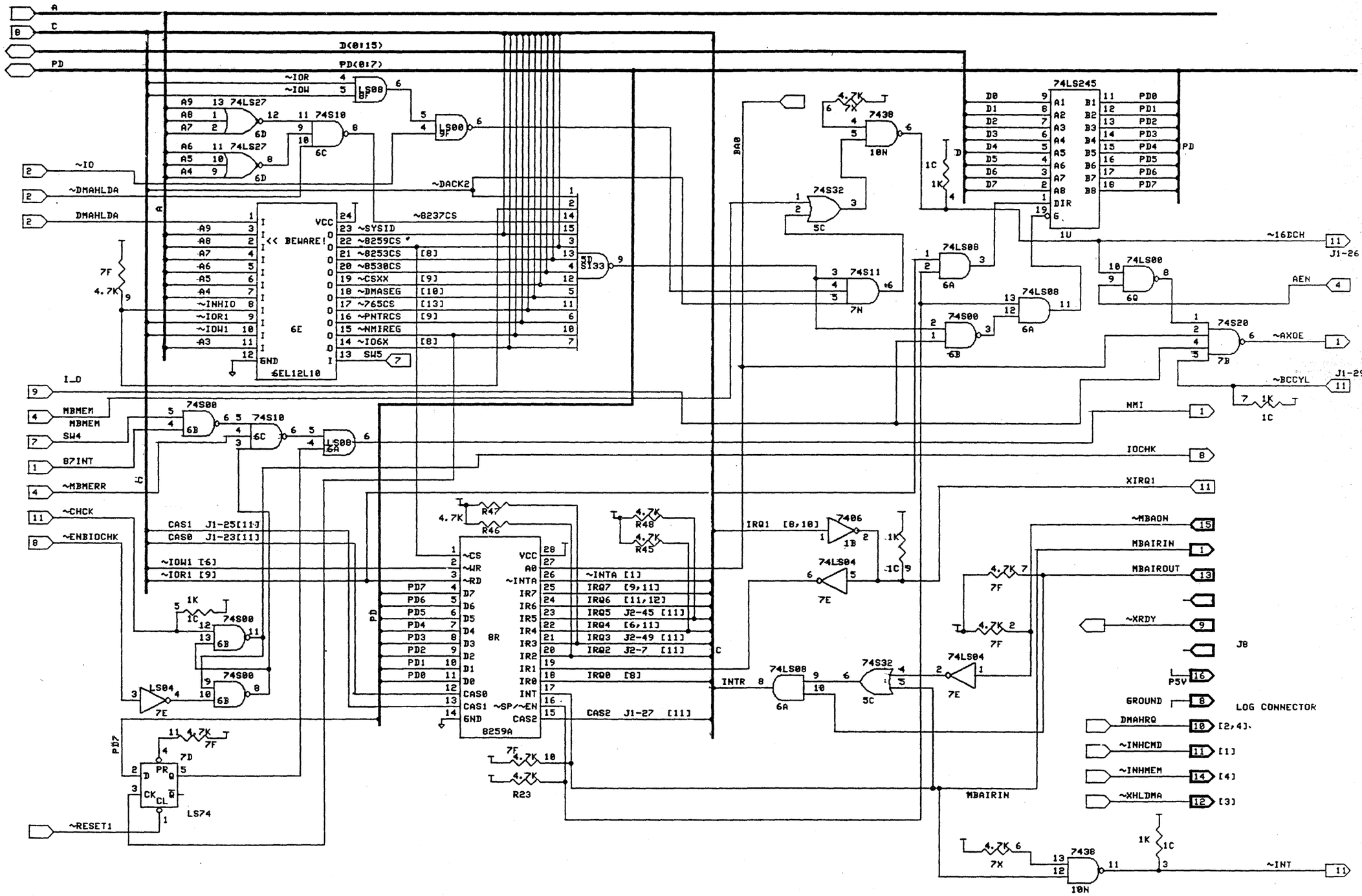
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
MOTHERBOARD
REV. P5 PAGE 8 OF 13



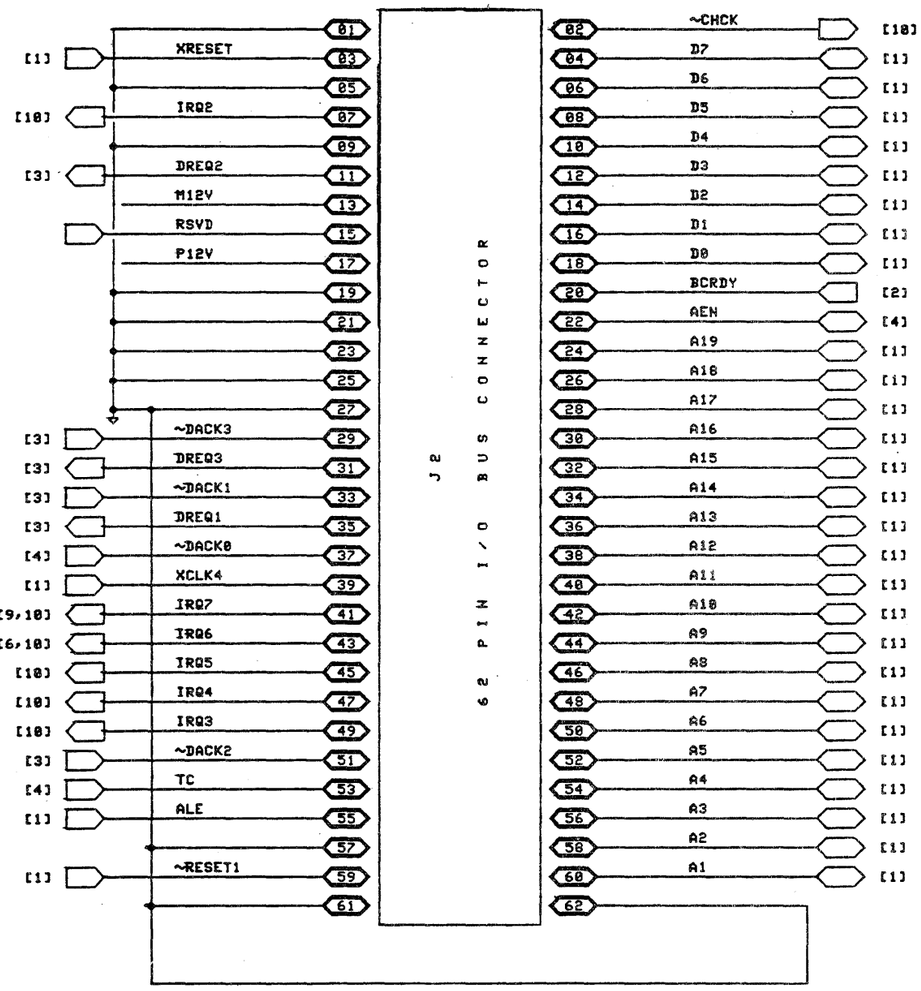
FOR TRAINING PURPOSES ONLY

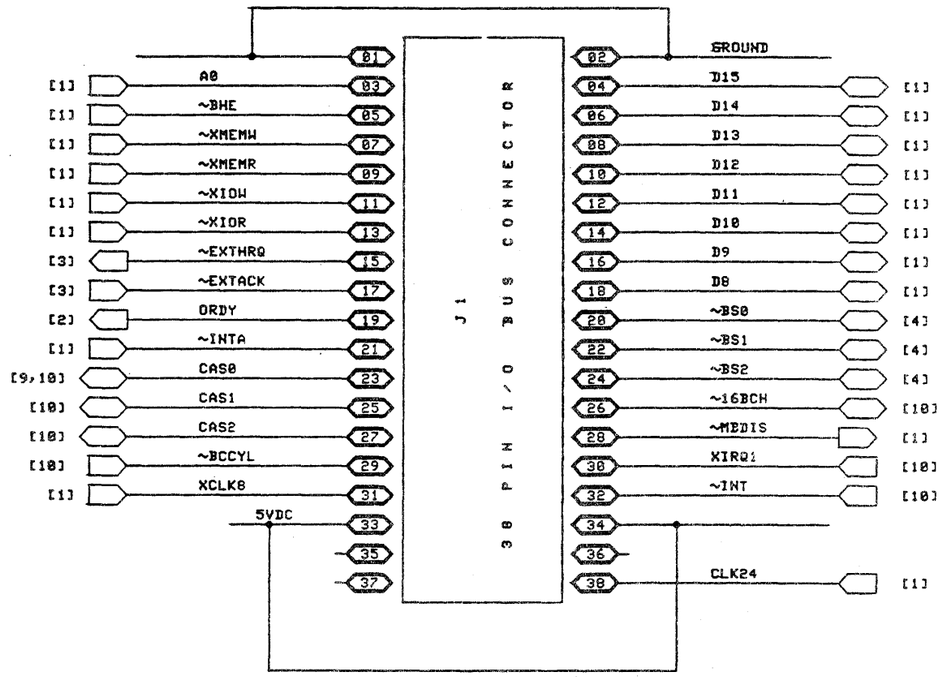
OLIVETTI PERSONAL COMPUTER M24
MOTHERBOARD
REV. P5 PAGE 9 OF 13



FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
MOTHERBOARD
REV. P5 PAGE 10 OF 13





PAGE BREAKDOWN

— 1] CPU

— 2] ARBITER AND WAIT LOGIC

— 3] DMA

— 4] MEMORY CONTROL, PROM, PARITY CKT

— 5] RAM

— 6] SERIAL COMMUNICATION CONTROLLER

— 7] CONFIGURATION SWITCH PORT

— 8] KEYBOARD CONTROLLER AND COUNTER TIMER.

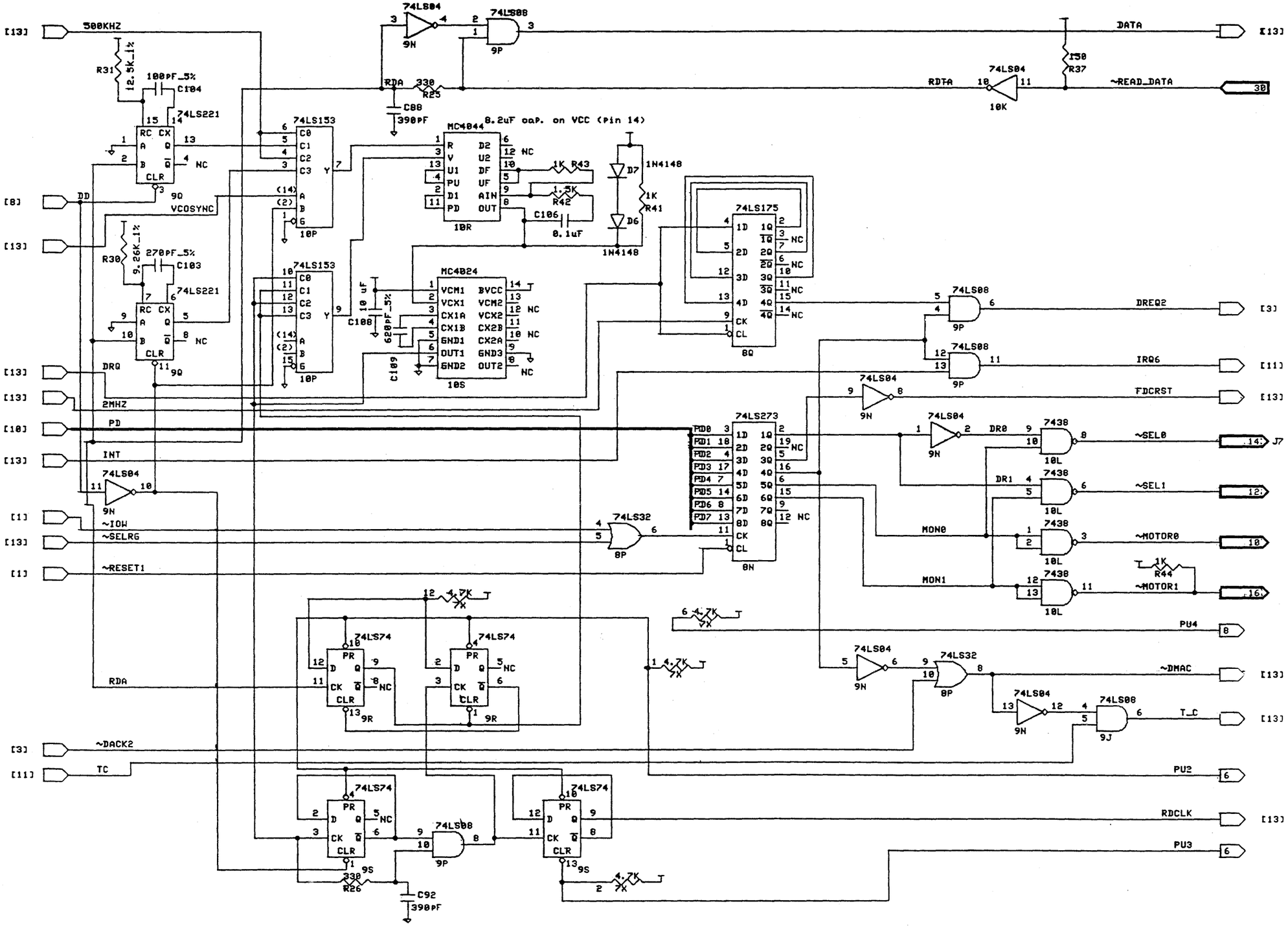
— 9] CLOCK CALENDAR CHIP & PRINTER INTERFACE

— 10] INTERRUPT CONTROLLER, IOCS, NMI LOGIC, ID PROM

— 11] (THIS PAGE) EXPANSION SOCKETS

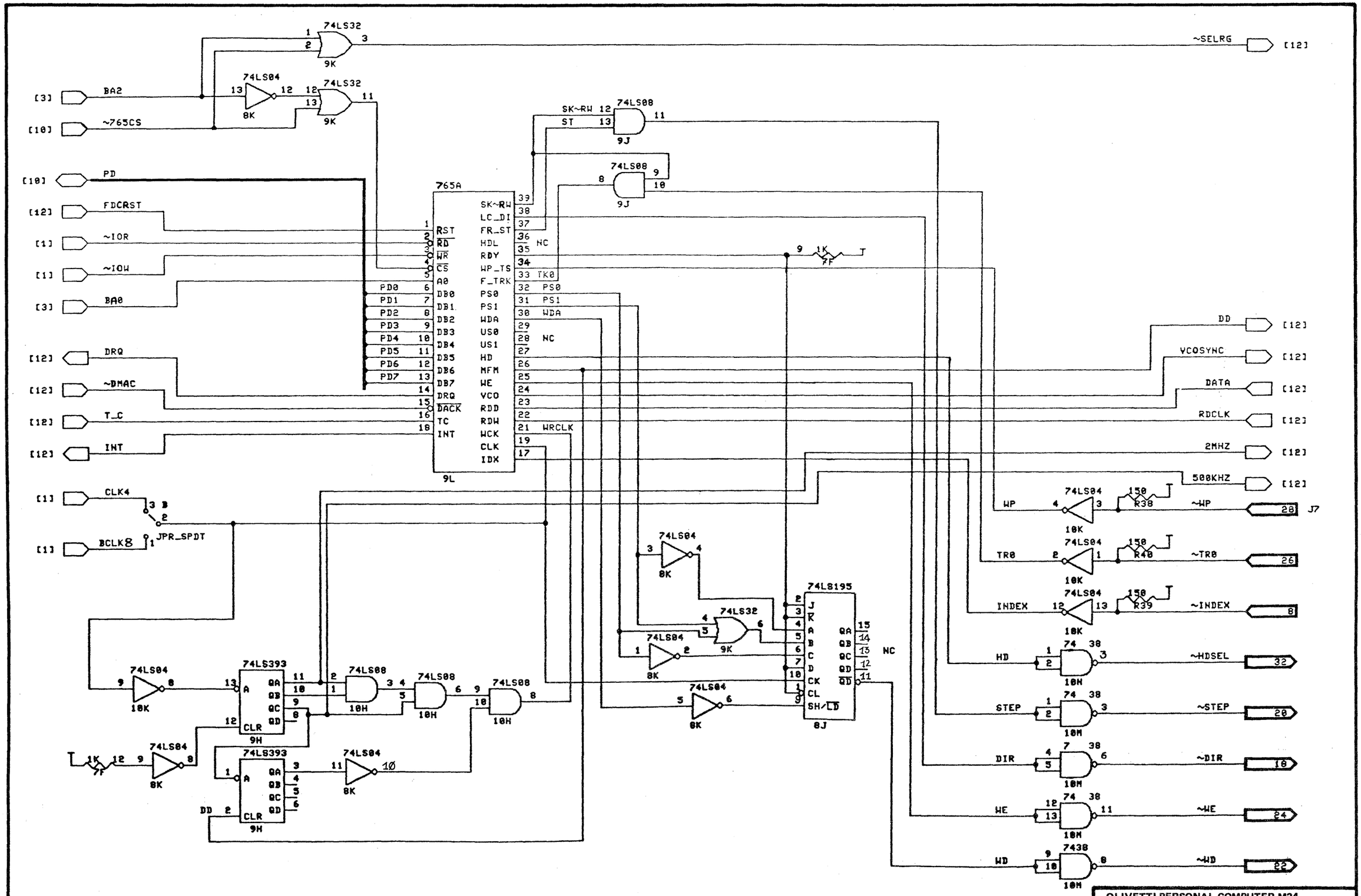
— 12] FLOPPY DISK CONTROLLER

— 13] FLOPPY DISK CONTROLLER



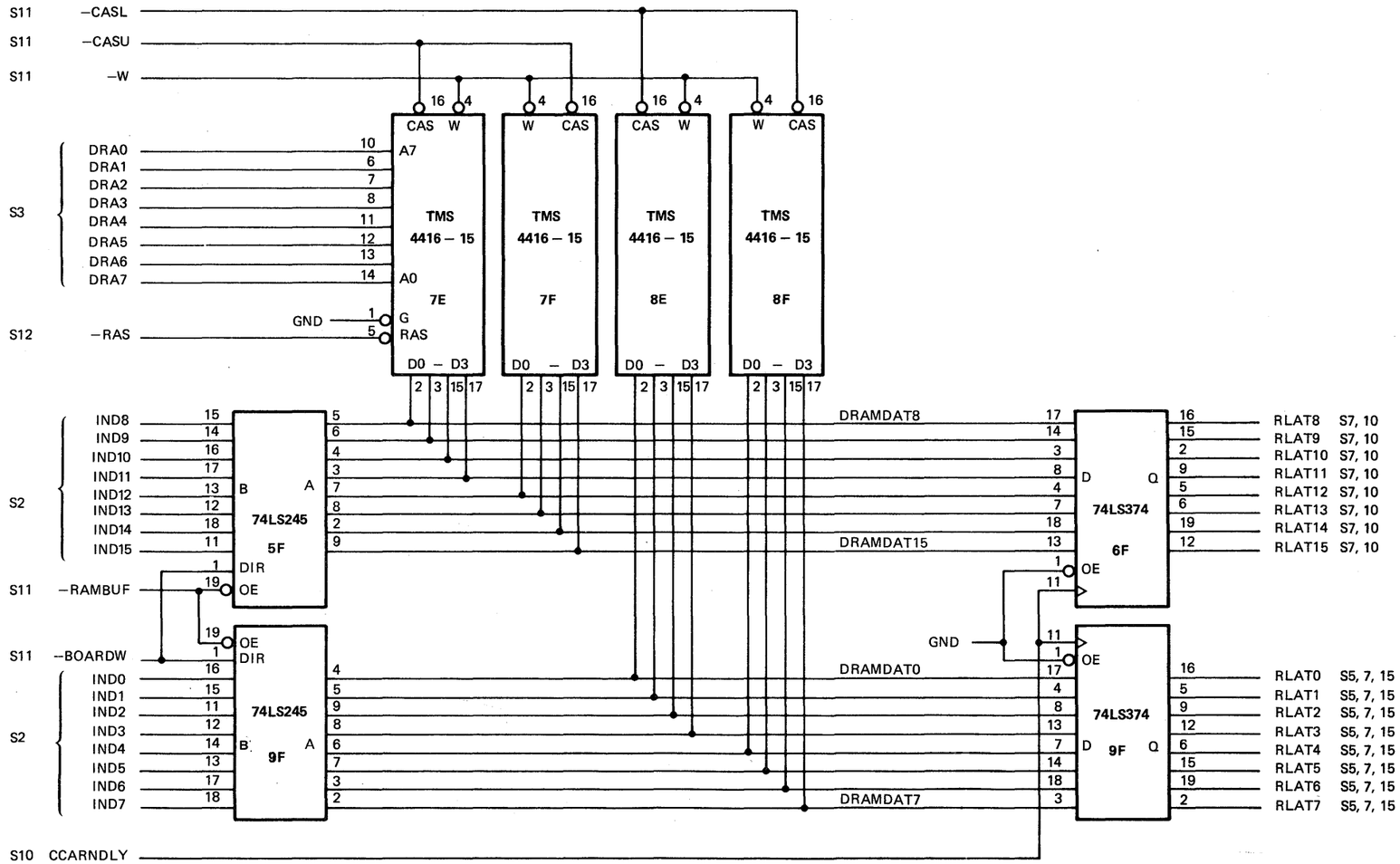
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
MOTHERBOARD
REV. P5 PAGE 12 OF 13



FOR TRAINING PURPOSES ONLY

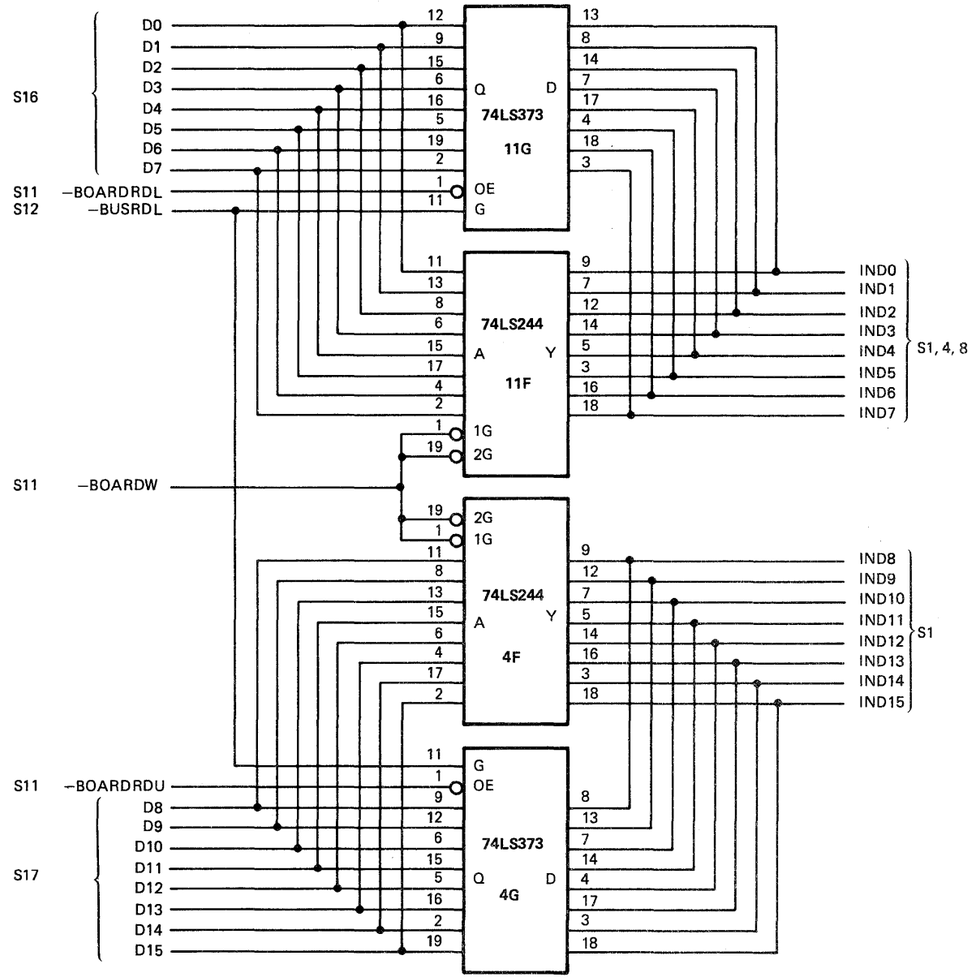
OLIVETTI PERSONAL COMPUTER M24
MOTHERBOARD
REV. P5 PAGE 13 OF 13



DISPLAY RAM

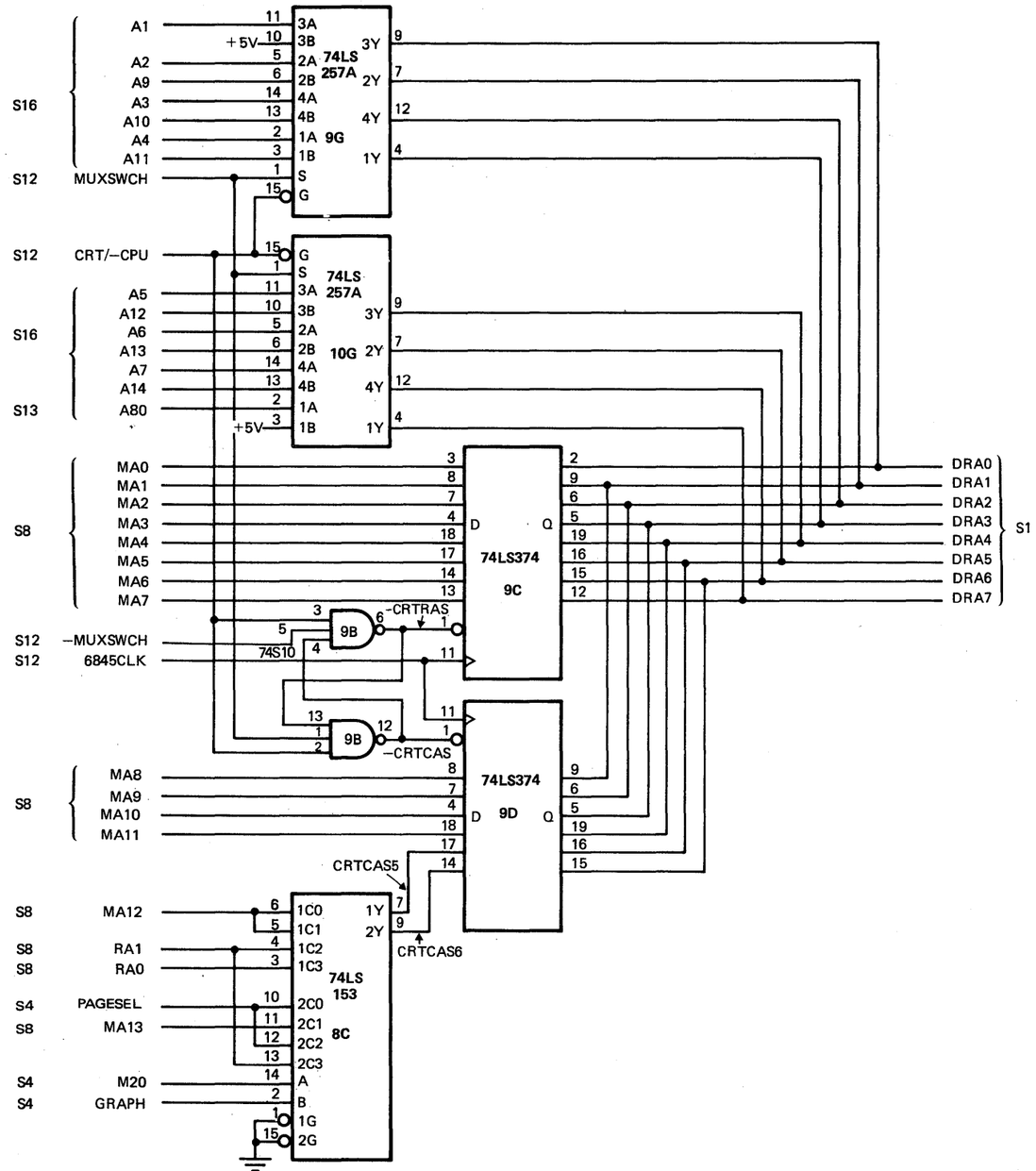
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
DISPLAY CONTROLLER
REV. P2 PAGE 1 OF 17



DATA BUS BUFFERING

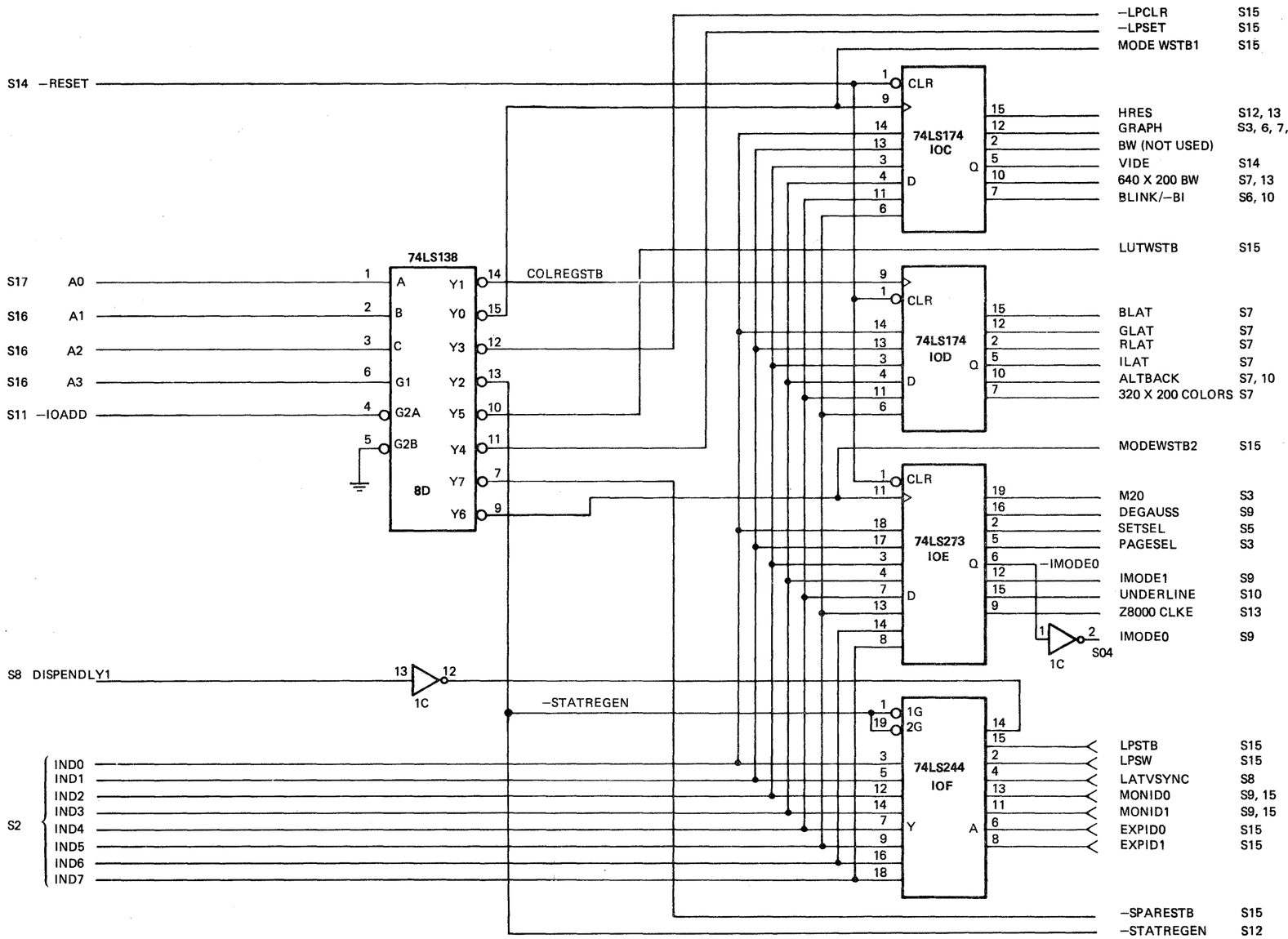
FOR TRAINING PURPOSES ONLY



DRAM ADDRESS MULTIPLEXER

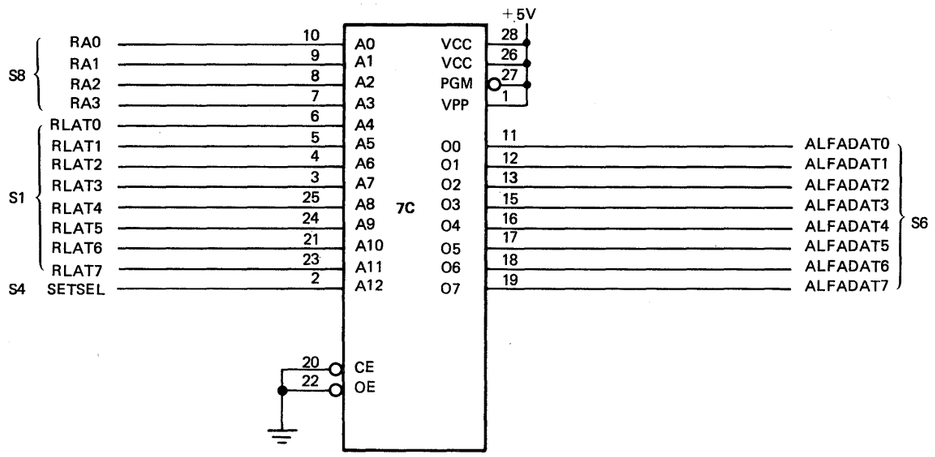
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
DISPLAY CONTROLLER
REV. P2 PAGE 3 OF 17



I/O REGISTERS

FOR TRAINING PURPOSES ONLY

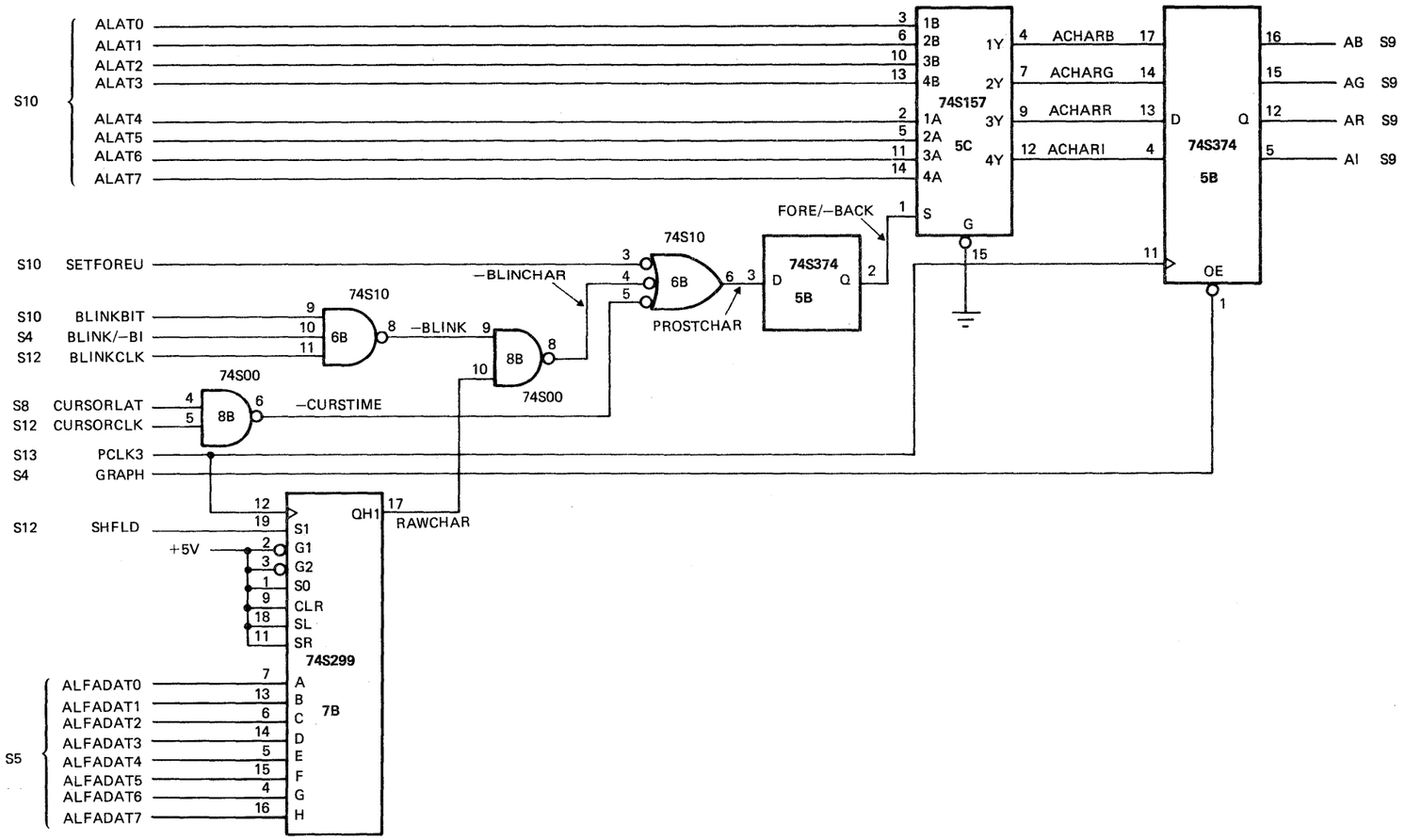


2732A EPROM (4K)
OR
2732A–25 EPROM (4K)
OR
2764 EPROM (8K)
OR
2764–25 EPROM (8K)
OR
SY2333–2 ROM (4K)
OR
SY2365–2 ROM (8K)
OR
SY2365A–2 ROM (8K)
OR
S2364B ROM (8K)

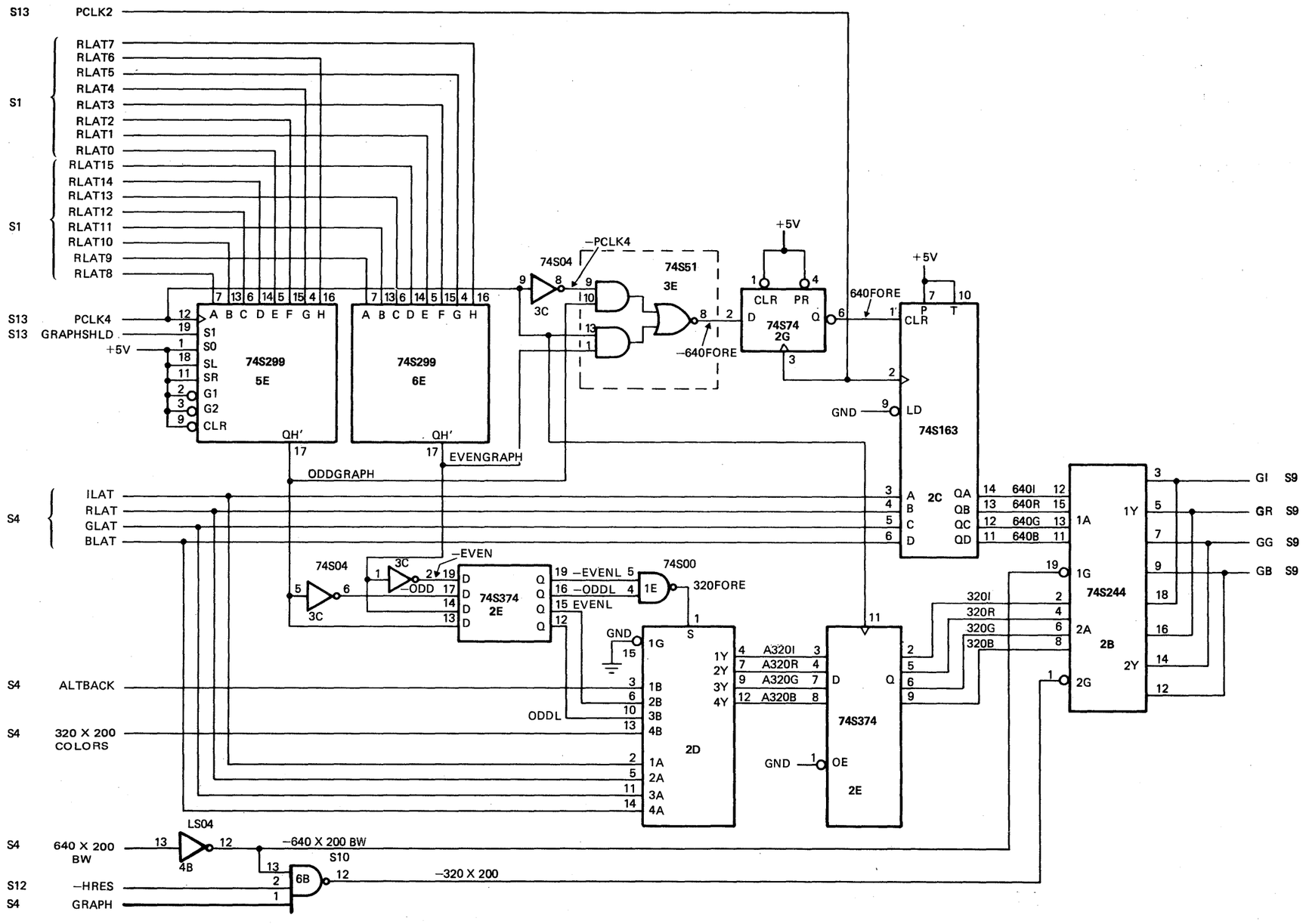
CHARACTER ROM

FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
DISPLAY CONTROLLER
REV. P2 PAGE 5 OF 17



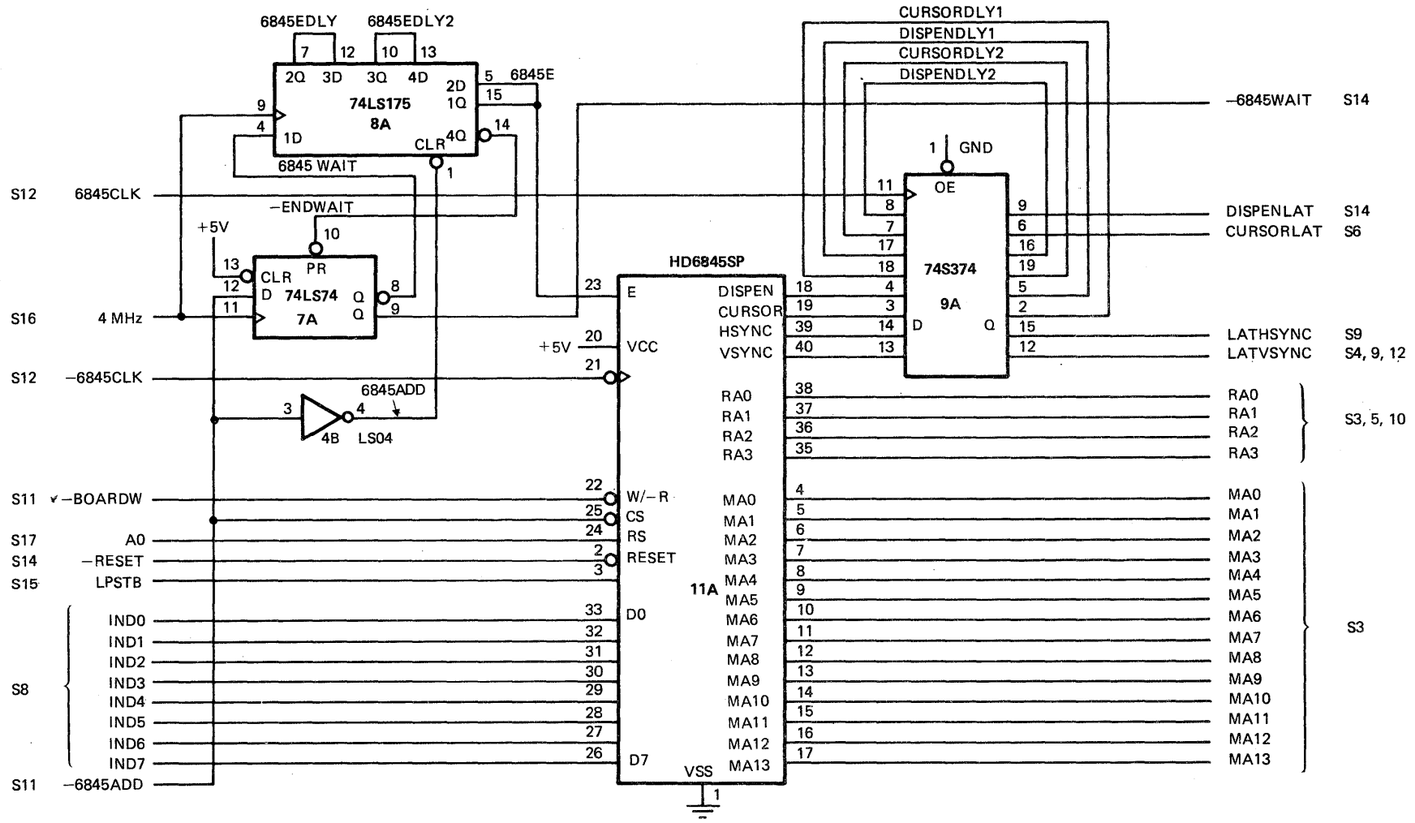
CHARACTER OUTPUT STAGE



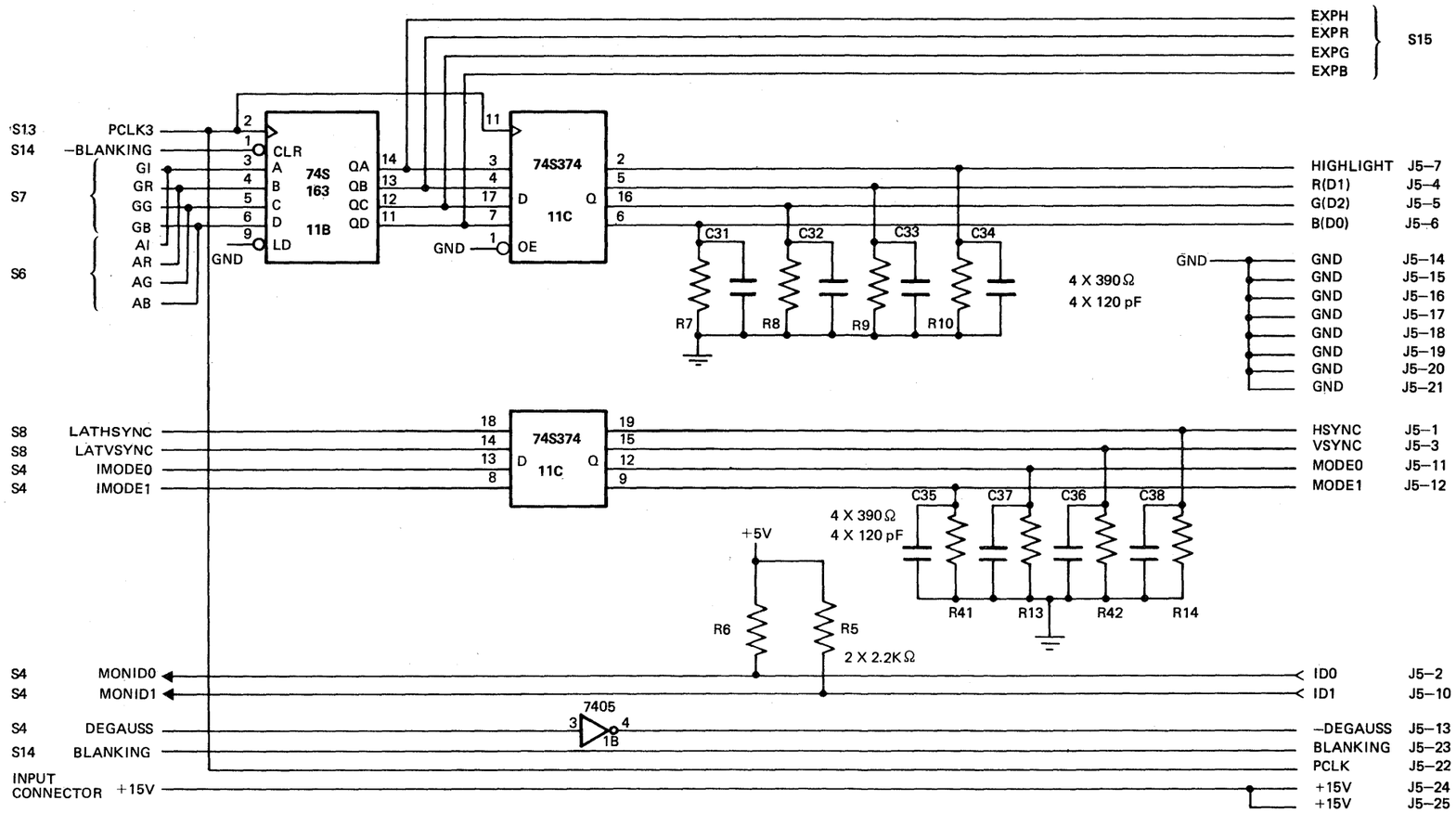
GRAPHICS OUTPUT STAGE

FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
DISPLAY CONTROLLER
REV. P2 PAGE 7 OF 17



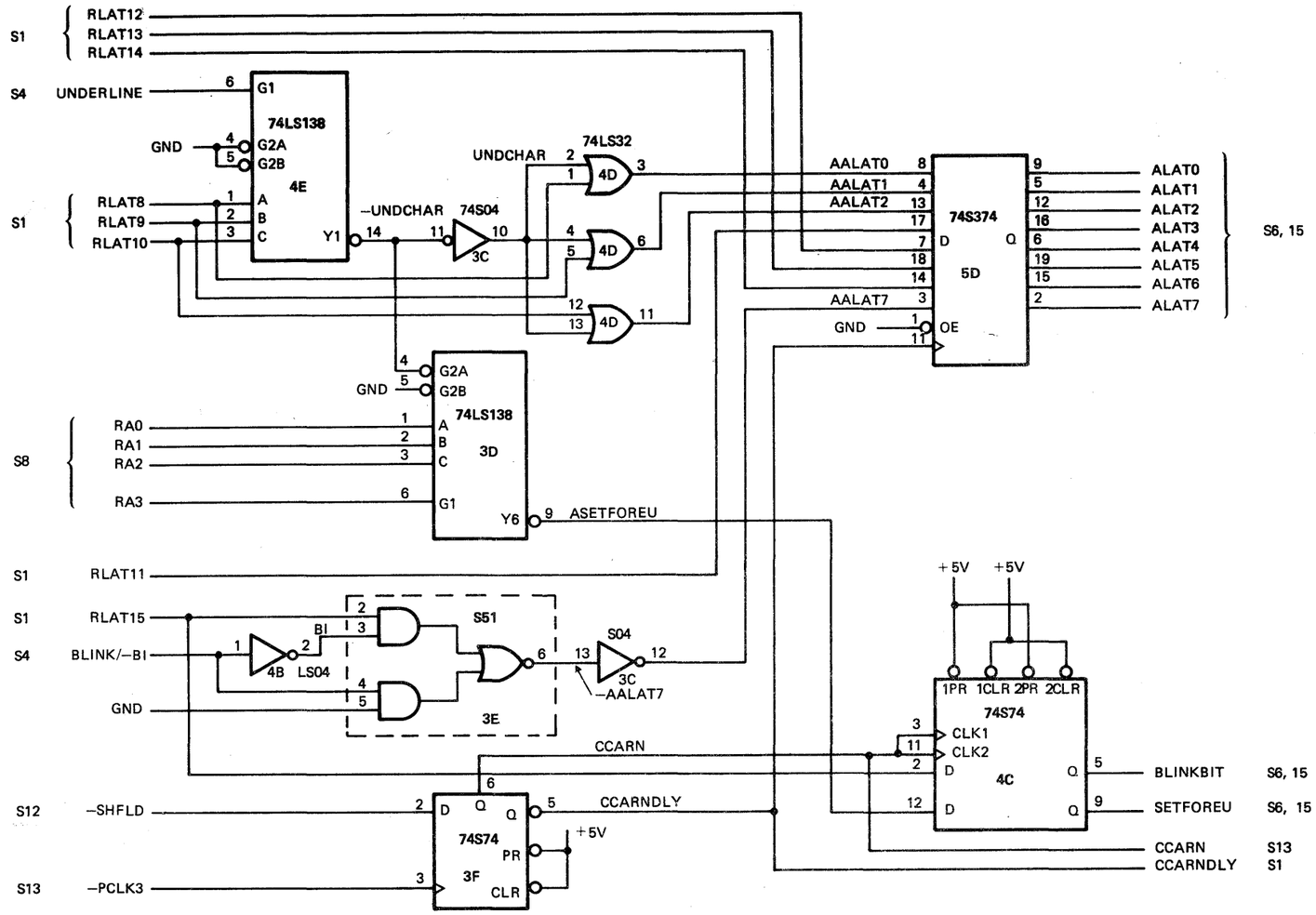
6845 CRT CONTROLLER



BLANKING AND OUTPUT DRIVER

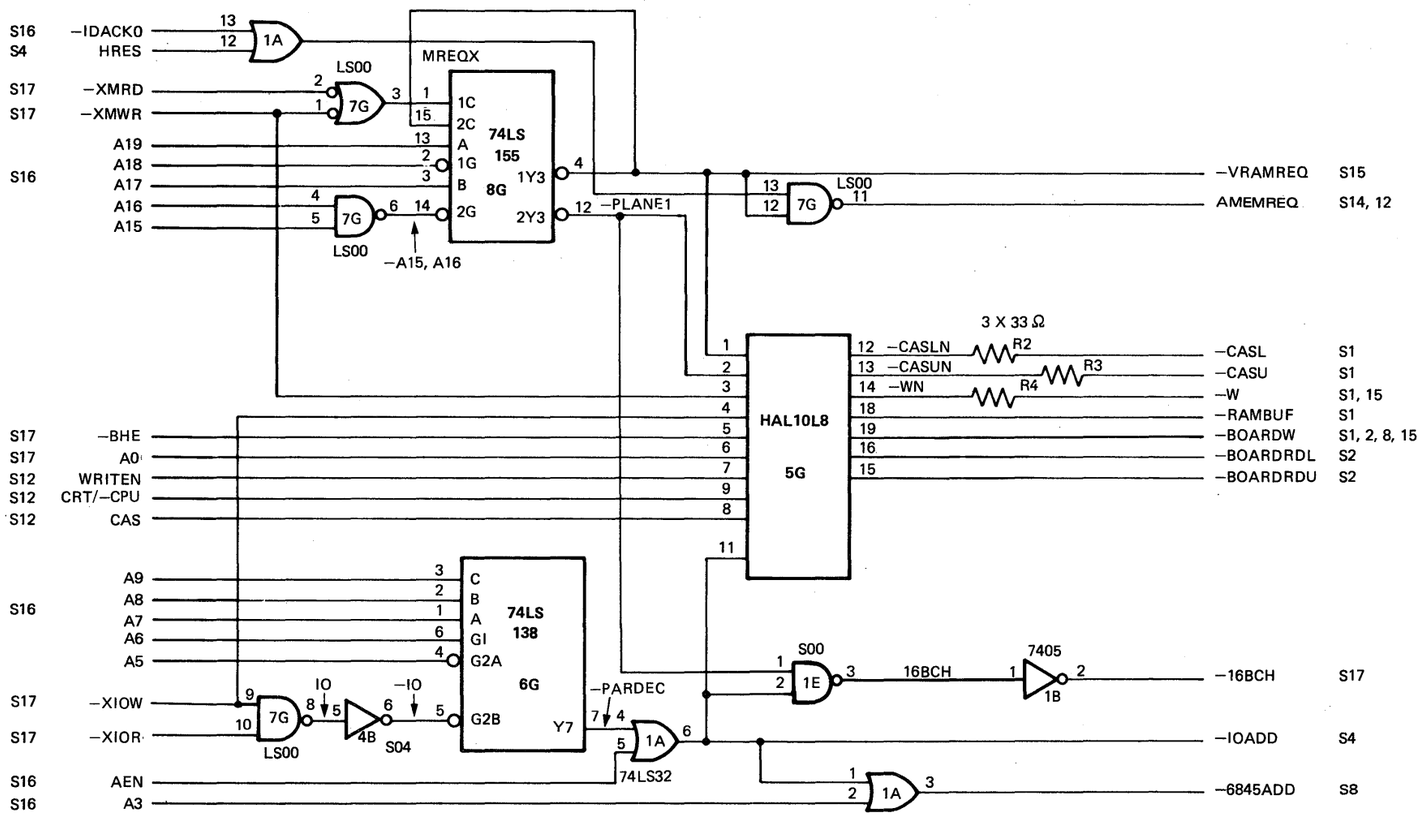
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
DISPLAY CONTROLLER
REV. P2 PAGE 9 OF 17



CHARACTER BLINK & UNDERLINE LOGIC

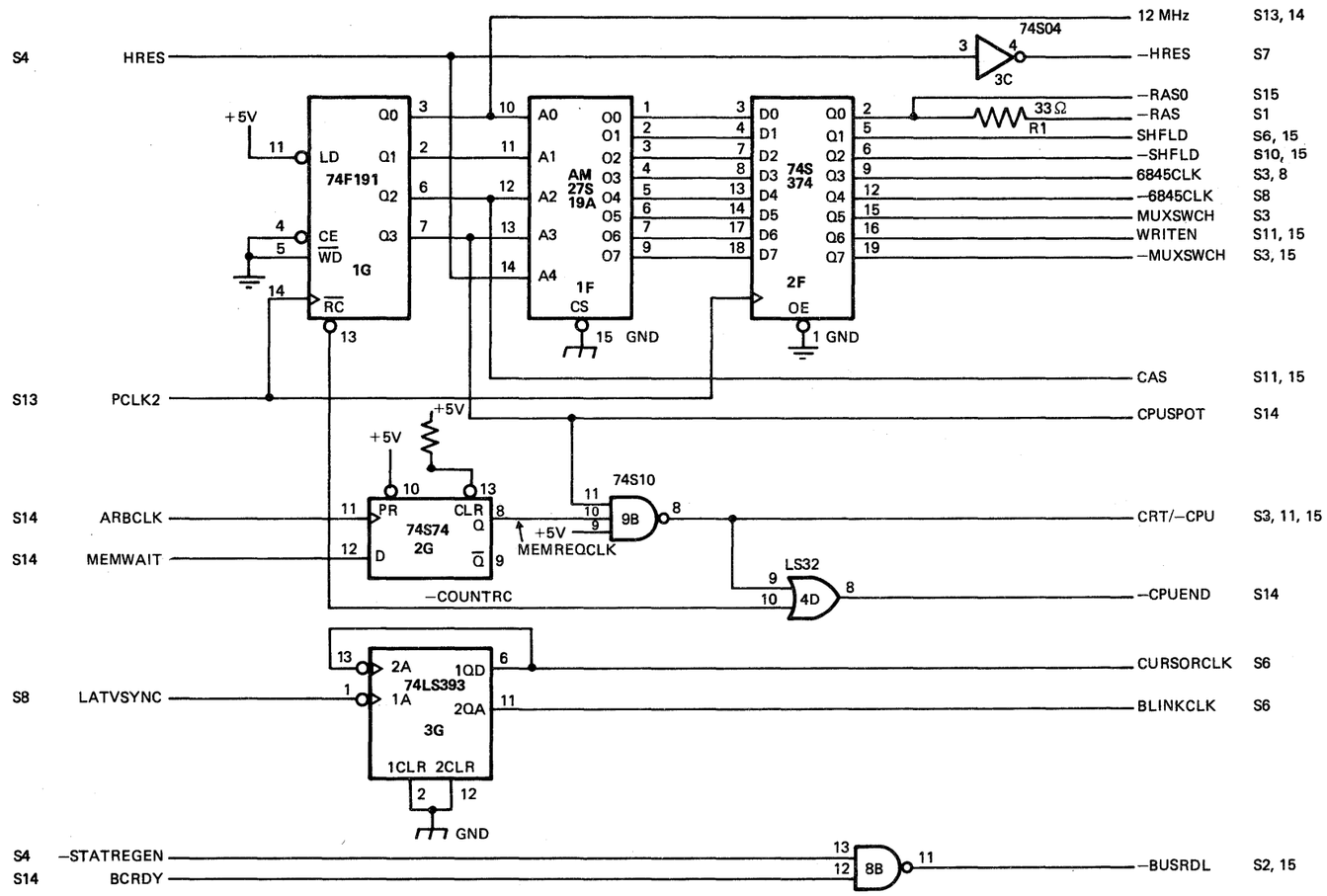
FOR TRAINING PURPOSES ONLY



ADDRESS DECODER

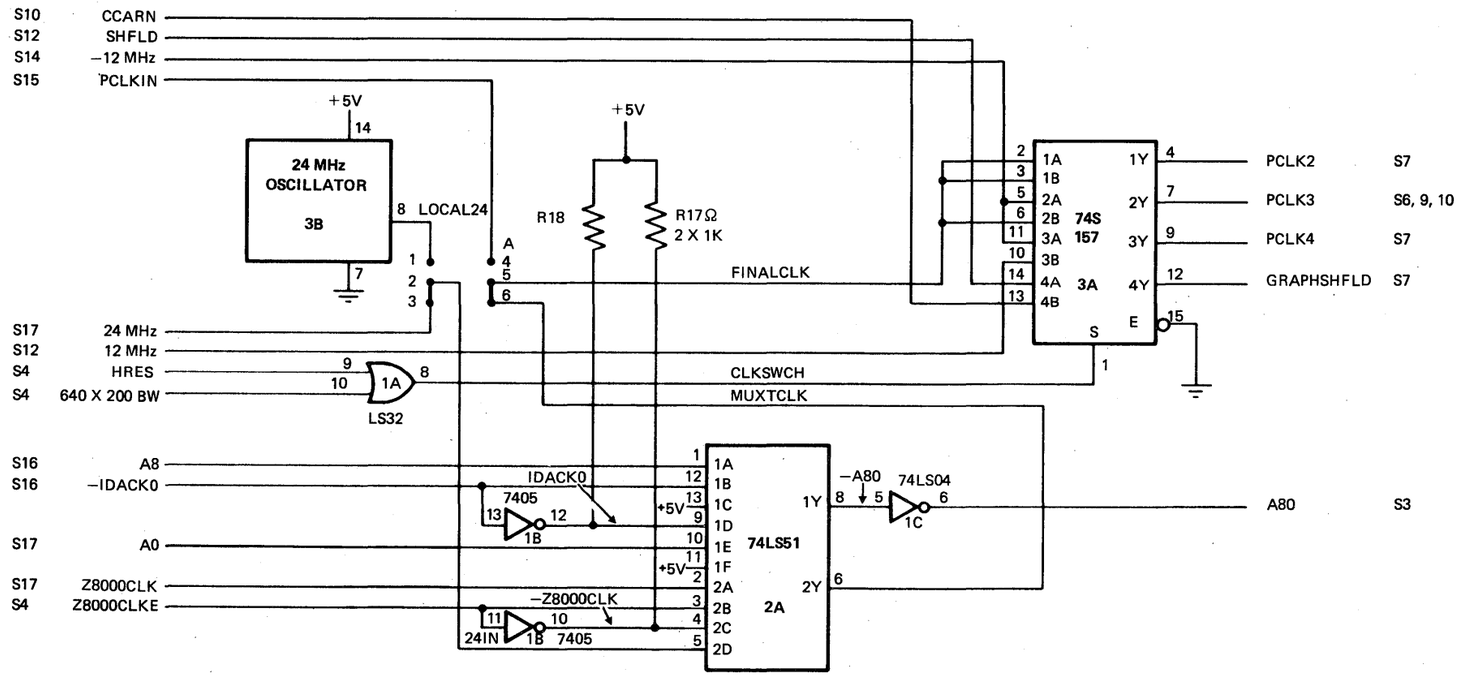
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
DISPLAY CONTROLLER
REV. P2 PAGE 11 OF 17



TIMING LOGIC

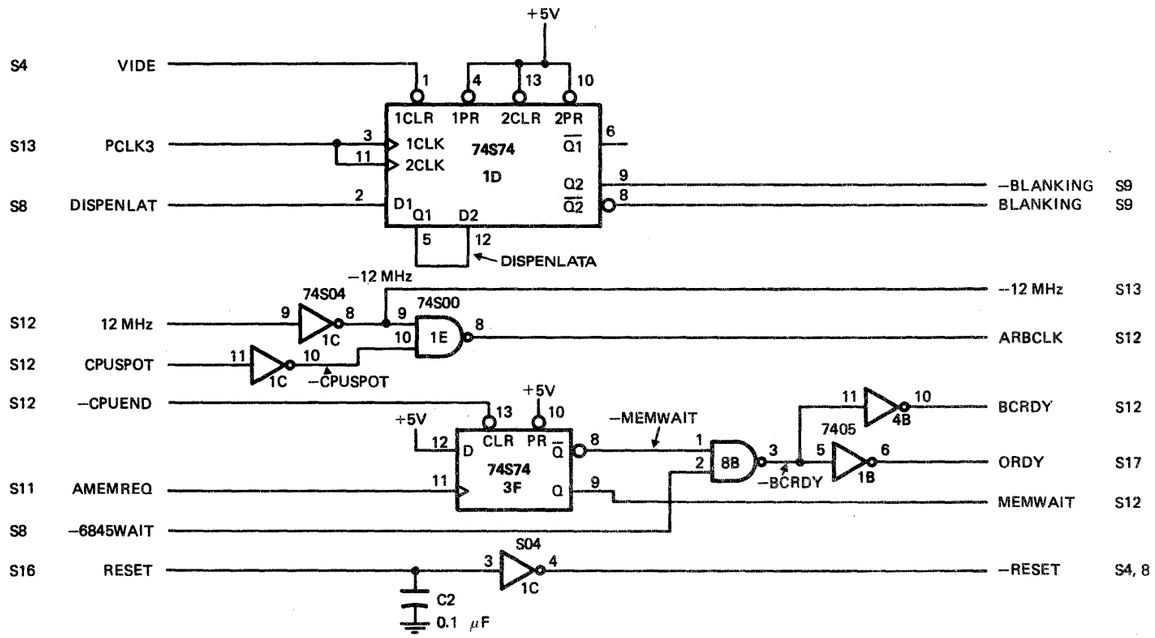
FOR TRAINING PURPOSES ONLY



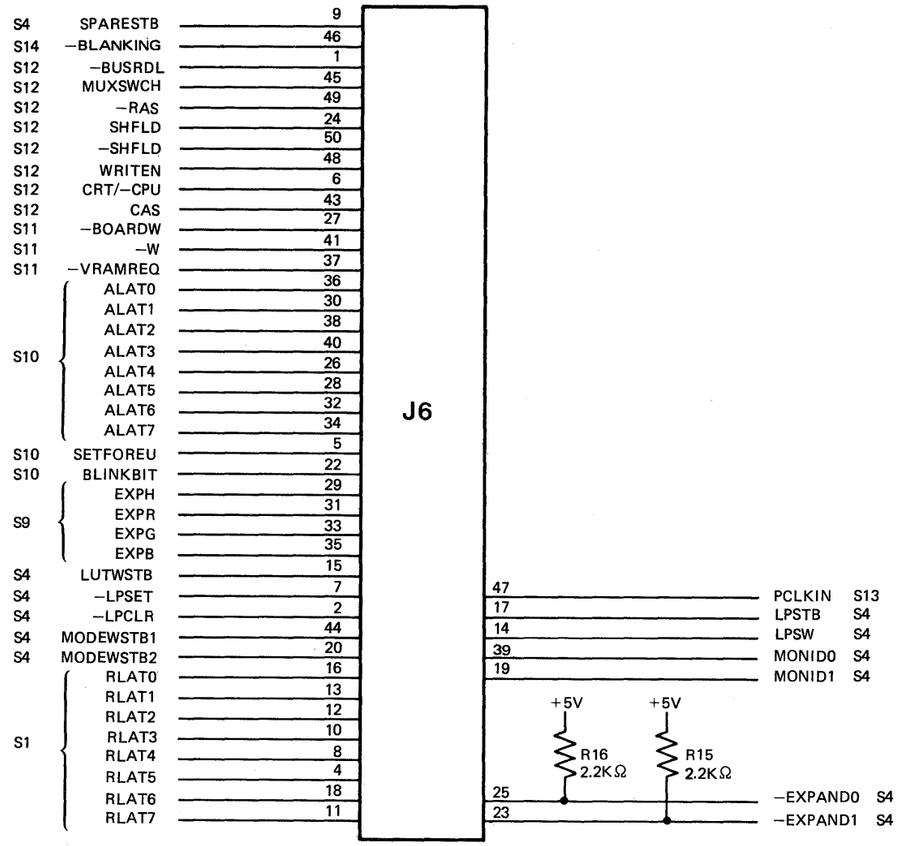
MODE CIRCUIT

FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
DISPLAY CONTROLLER
REV. P2 PAGE 13 OF 17



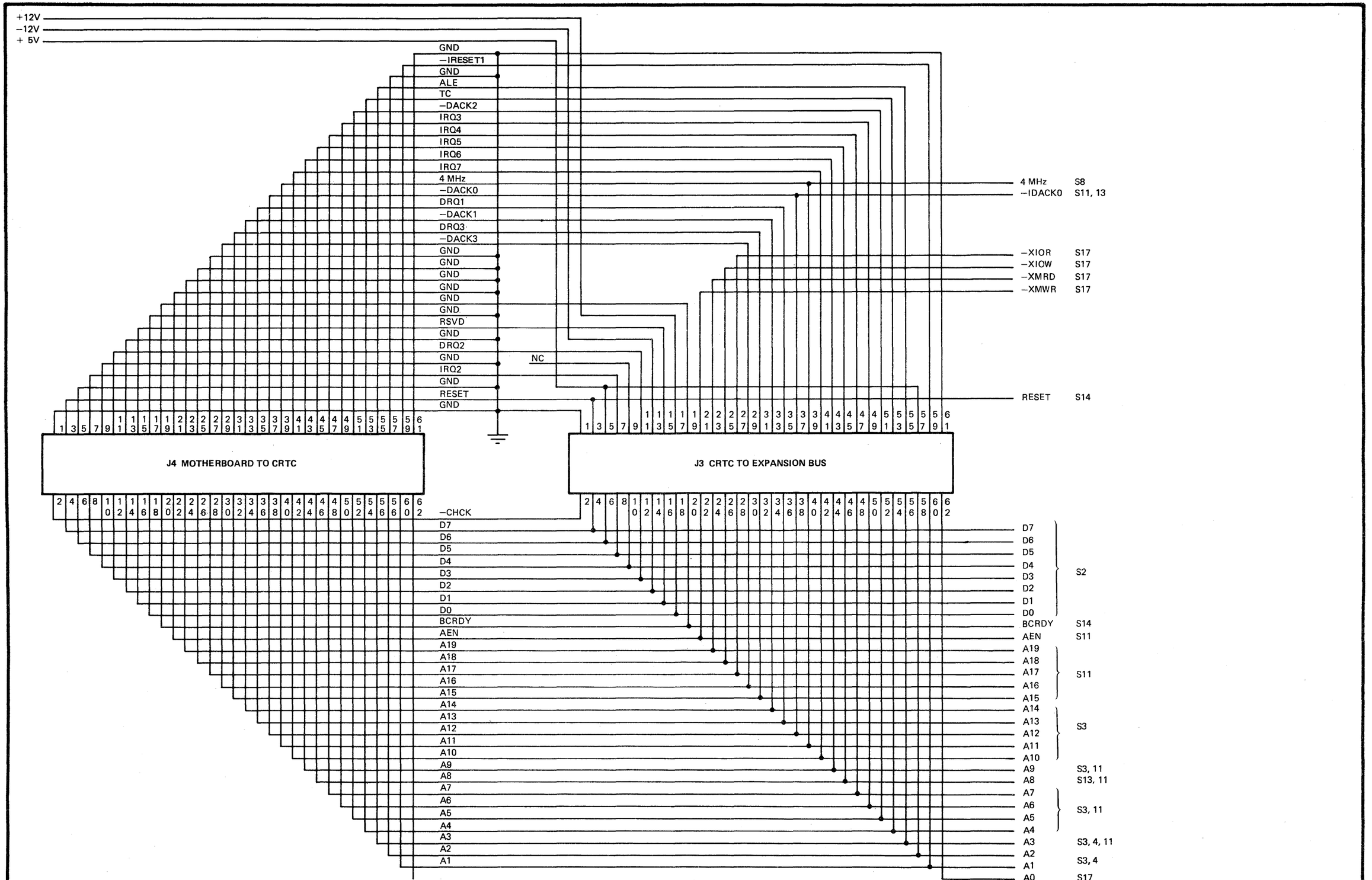
BLANKING, ARBITRATION AND READY PROCESSING



ENHANCEMENT CONNECTOR

FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
DISPLAY CONTROLLER
REV. P2 PAGE 15 OF 17



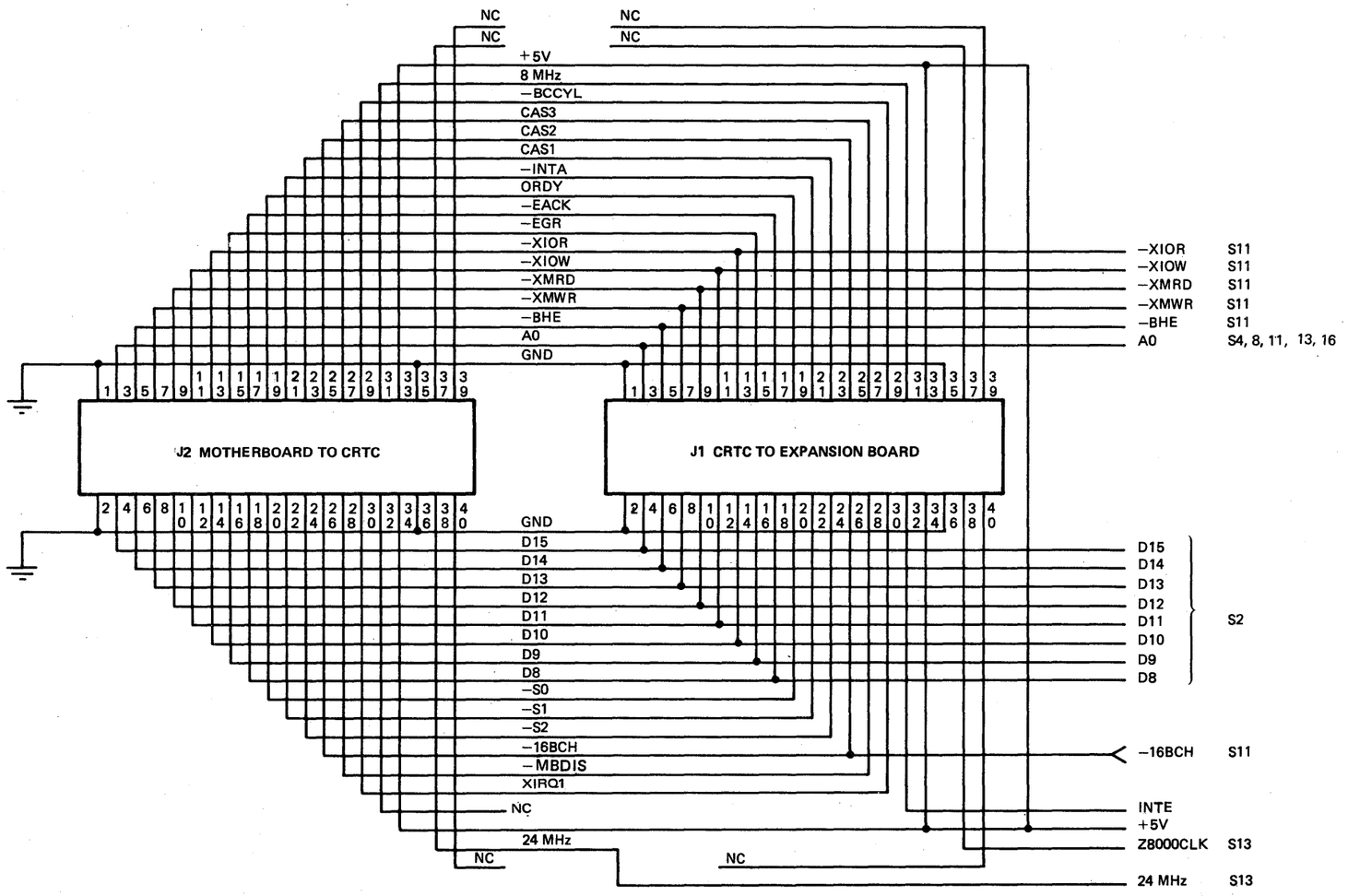
FOR TRAINING PURPOSES ONLY

SYSTEM CONNECTORS

OLIVETTI PERSONAL COMPUTER M24

DISPLAY CONTROLLER

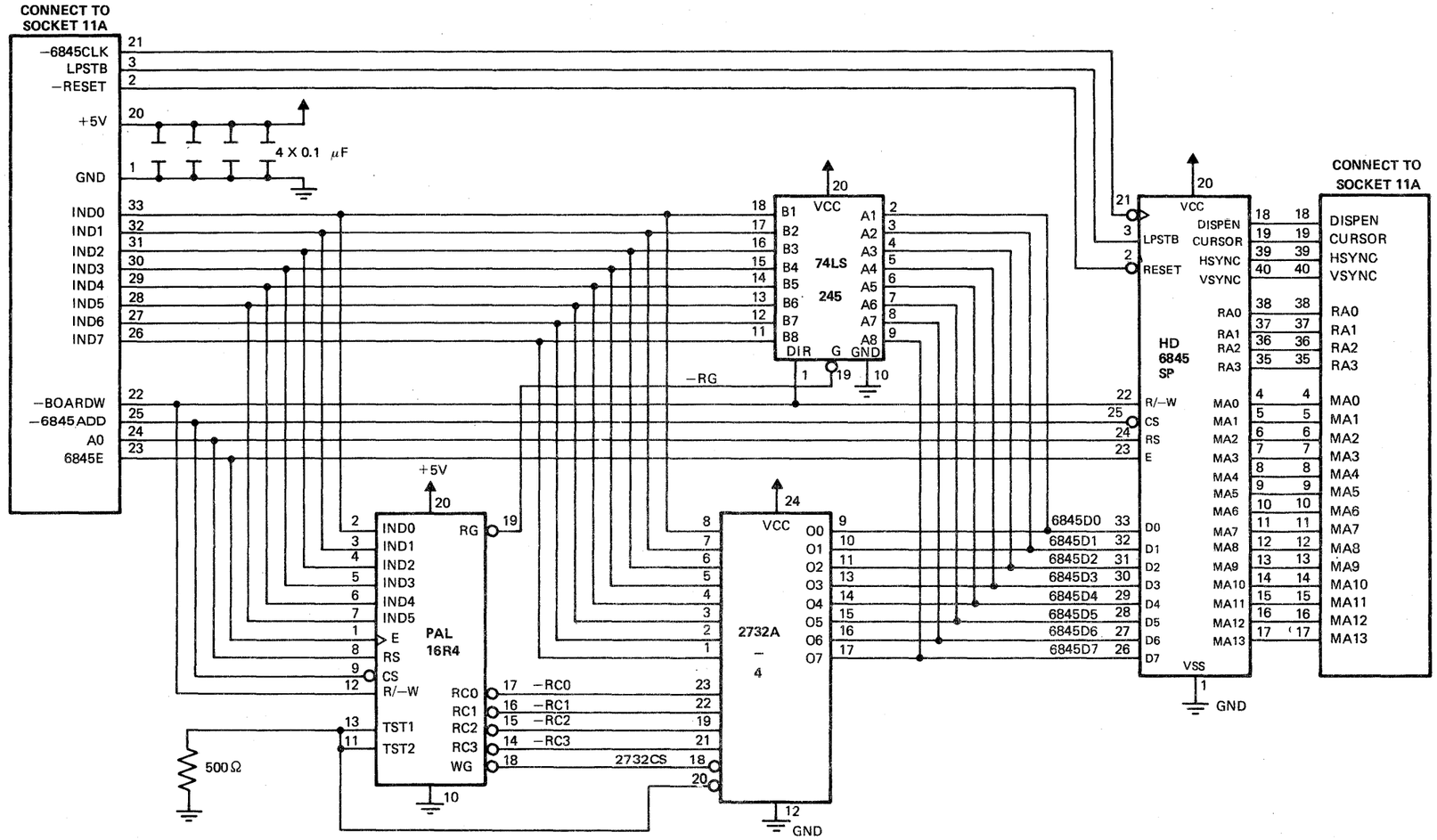
REV. P2 PAGE 16 OF 17



SYSTEM CONNECTORS

FOR TRAINING PURPOSES ONLY

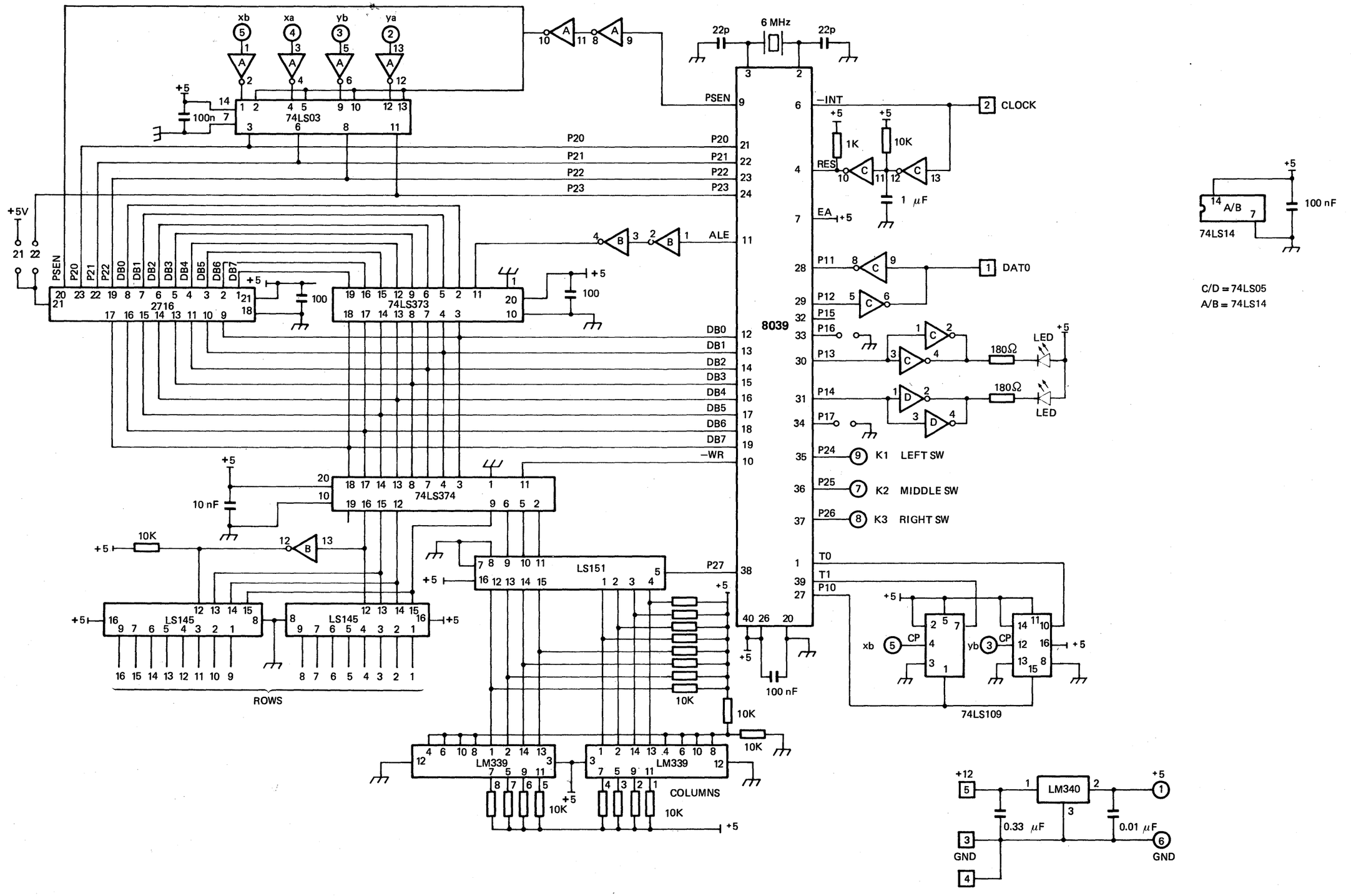
OLIVETTI PERSONAL COMPUTER M24
DISPLAY CONTROLLER
REV. P2 PAGE 17 OF 17



SCRAMBLER BOARD

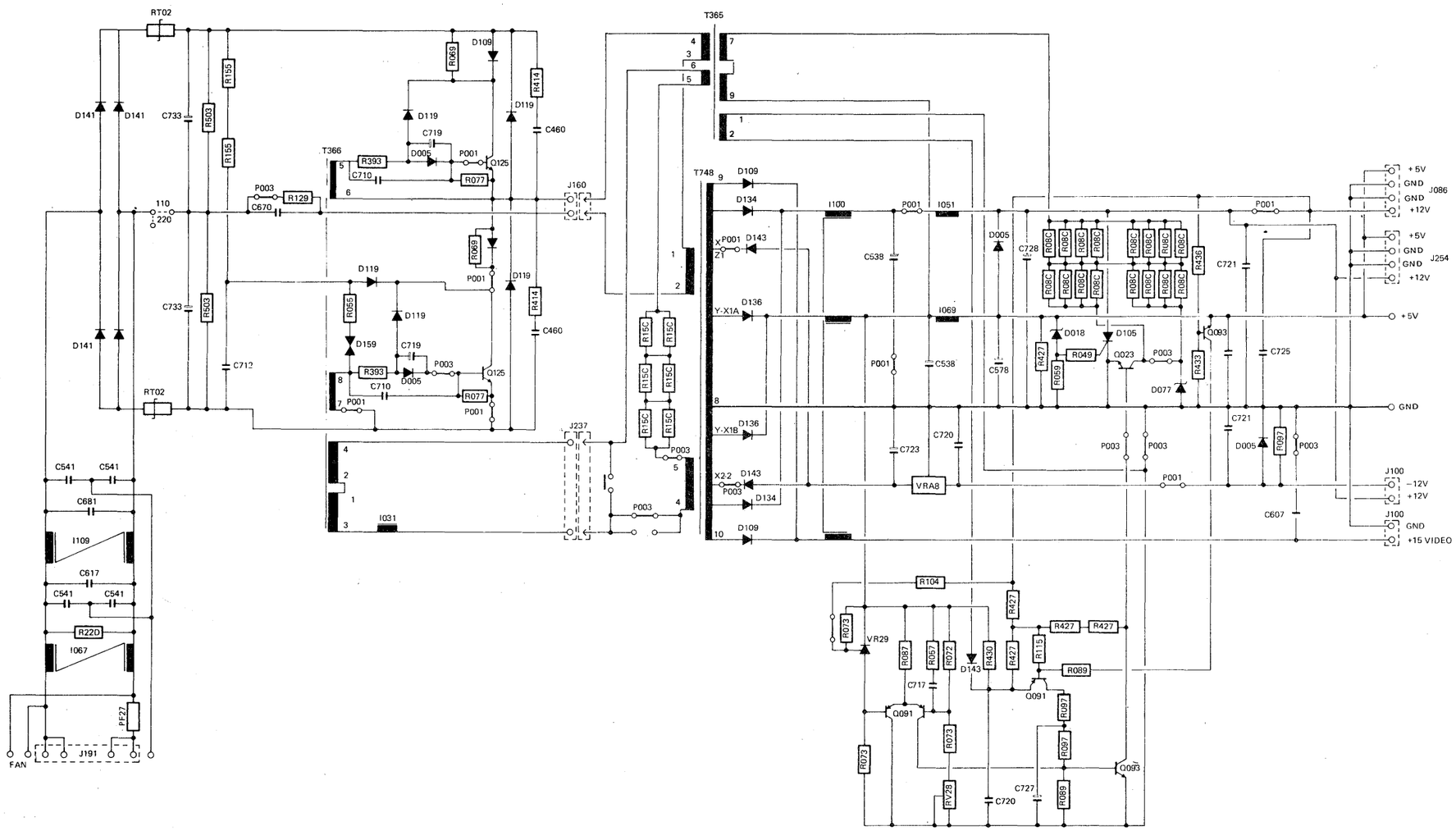
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
SCRAMBLER BOARD
PAGE 1 OF 1



FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
KEYBOARD
PAGE 1 OF 1

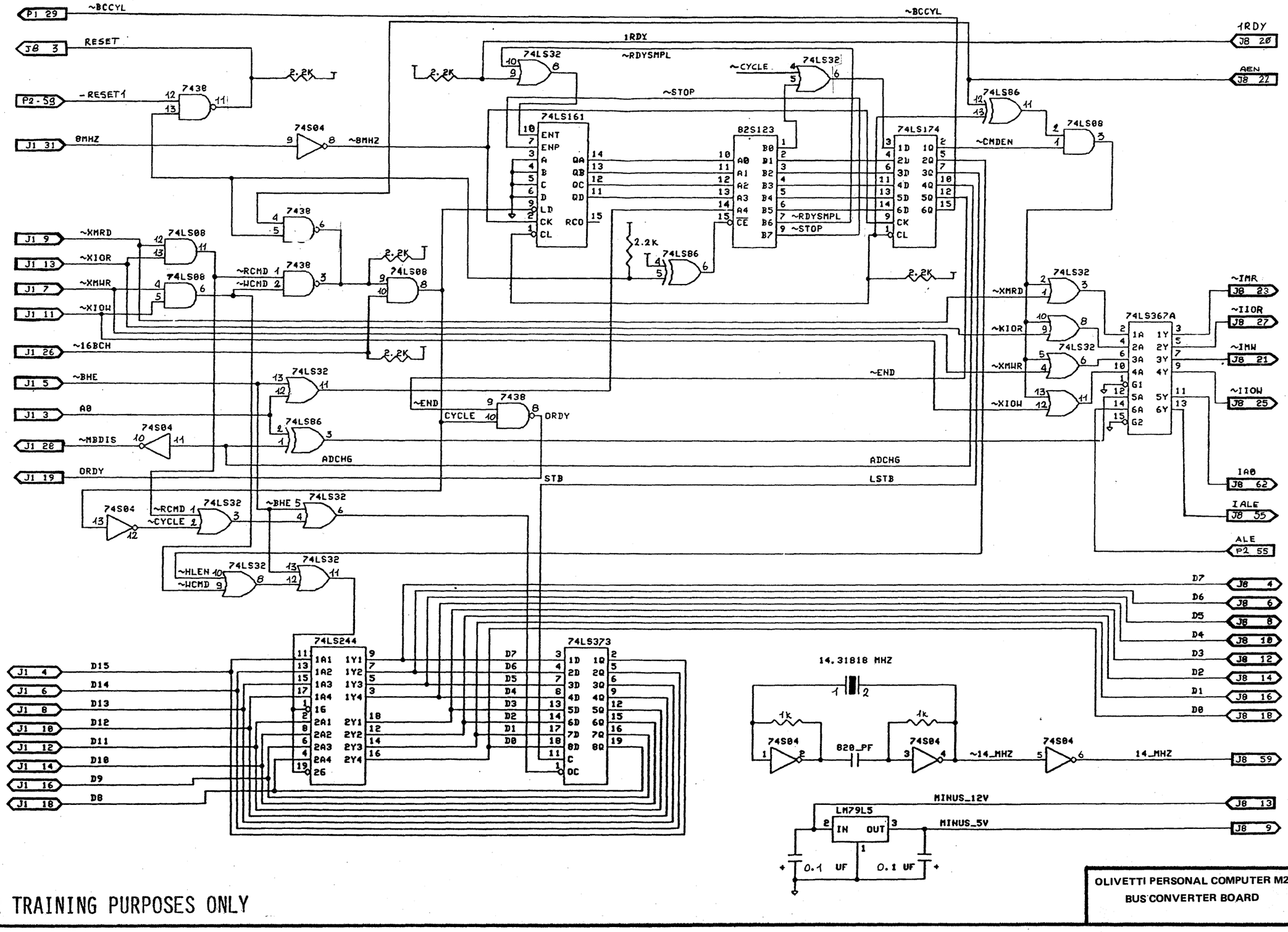


K	KLIXON
VR29	TL 431C
VRA8	REG. −12V 1A
T438	TRANSF. T438
T366	TRANSF. T366
T365	TRANSF. T365
RT02	THERMIS. 82Ω 20%
PF27	FUSE 3.15A 250V RIT. BPR
P003	P7.62 JUMPER
P001	P15.24 JUMPER
J254	CONN. M 90° MOLEX 4 WAY
J237	CONN. M DIR. 2 WAY MODU II
J191	CONN. M DIR. 4 WAY MOLEX 5,08
J100	CONN. M 90° 2 WAY MODU I
J086	CONN. M 90° 4 WAY MODU II
J060	CONN. M DIR. 2 WAY MODU I
I109	IND. I109
I069	IND. I069
I100	IND. I100 MULTIPLE
I067	IND. I067
I051	IND. I051
I031	IND. I031
Q125	TRANS. BUV 47
Q093	TRANS. BC 237/B
Q091	TRANS. BC 307/B
Q023	TRANS. 4A 60V D44C8

Q054	TRANS. BC237/B	
Q053	TRANS. BC307/B	
C733		COND. 680 μF 200V ALL.
C527	4.7 μF 25V	C717 COND. 4,7 μF 25V ALL. N
C730	1000 μF 16V	C728 COND. 1000 μF 16V EL. N
C498	470 μF 16V	C727 COND. 470 μF 16V EL. N
C526	1000 μF 6.3V	C725 COND. 1000 μF 6.3V EL. N
C208	200 μF 25V	C723 COND. 220 μF 25V EL. N
C613	0,47 μF 50/63V	C721 COND. 0,47 μF 50/63V FILM N
C508	100 μF 25V	C720 COND. 100 μF 25V EL. N
C480	22 μF 25V	C719 COND. 22 μF 25V EL. N
C235	0,1 μF 50/63V	C712 COND. 0,1 μF 50/63V FILM N
C237	1 μF 50/63V	C710 COND. 1 μF 50V CER. N
		C681 COND. 0,1 μF 250V POLIPR.
		C670 COND. 6,8 μF 63V
		C617 COND. 0,47 μF 250V SCAT.
		C607 COND. 2200 μF 25V AL. VERT.
		C583 COND. 10000 μF 6,3V
		C578 COND. 2200 μF 10V AL. VERT.
		C541 COND. 3300 pF 250V
		C538 COND. 4700 μF 16V
		C460 COND. 470 pF 630V POLIPR.
		D159 DIAC BR 100 36V 2A
		D143 DIODE 100V 1A IN4934
		D141 DIODE 600V 3A MR 506
		D136 DIODE FAST. BYW 31
		D134 DIODE FAST. 12A 100V BYW 30
		D119 DIODE RAP. 1A 600V IN4937
		D109 DIODE 100V 3A MR 851
		D105 DIODE SCR C122B 200V 8A
		D077 DIODE ZENER 24V 5W IN5359 B
		D018 DIODE ZENER 5,6V 0,4W IN752 A
		D005 DIODE 100V 1A IN4002
		R104 RESIST. 2KΩ 1/4W 5%
		RV28 POTENT. 50Ω 3/4W
		R22D RESIST. 560KΩ 1/2W 5%
		R15C RESIST. 20Ω 1/2W 5%
		R08C RESIST. 10Ω 1/2W 5%
		R503 RESIST. 150KΩ 1/2W 2%
		R436 RESIST. 240Ω 1/2W 2%
		R433 RESIST. 180Ω 1/2W 2%
		R430 RESIST. 130Ω 1/2W 2%
		R427 RESIST. 100Ω 1/2W 2%
		R414 RESIST. 30Ω 1/2W 2%
		R393 RESIST. 3,9Ω 1/2W 2%
		R155 RESIST. 270KΩ 1/4W 5%
		R139 RESIST. 56KΩ 1/4W 5%
		R115 RESIST. 5,6KΩ 1/4W 5%
		R097 RESIST. 1KΩ 1/4W 5%
		R089 RESIST. 470Ω 1/4W 5%
		R081 RESIST. 220Ω 1/4W 5%
		R077 RESIST. 150Ω 1/4W 5%
		R073 RESIST. 100Ω 1/4W 5%
		R072 RESIST. 91Ω 1/4W 5%
		R069 RESIST. 68Ω 1/4W 5%
		R059 RESIST. 27Ω 1/4W 5%
		R057 RESIST. 22Ω 1/4W 5%
		R055 RESIST. 18Ω 1/4W 5%
		R049 RESIST. 10Ω 1/4W 5%

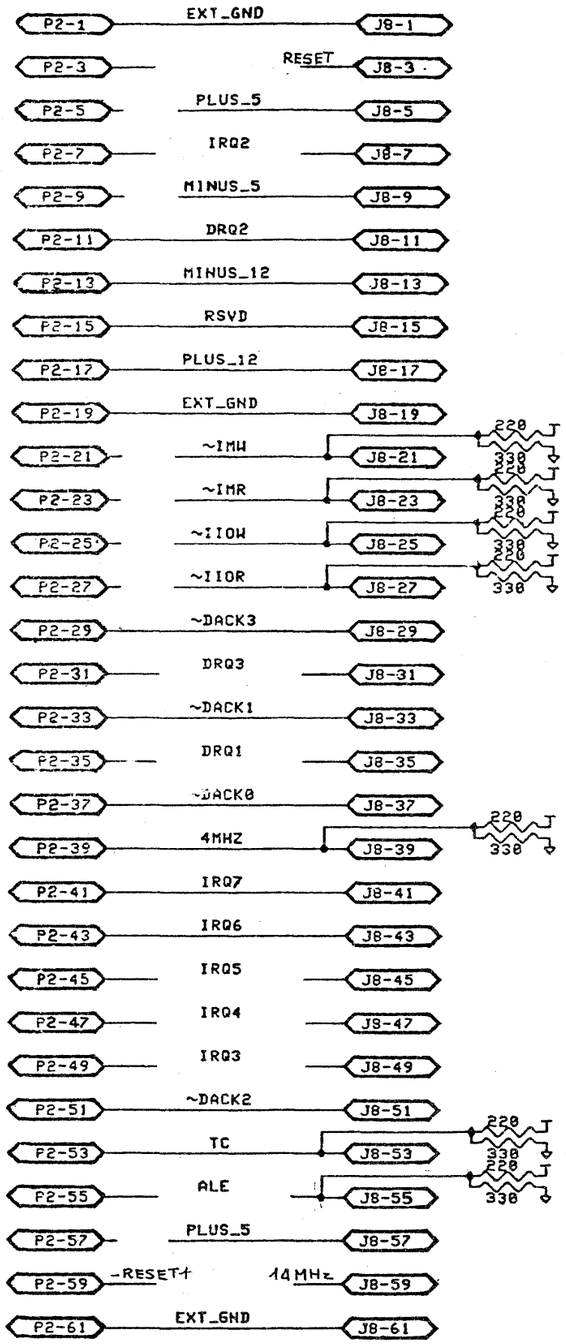
FOR TRAINING PURPOSES ONLY

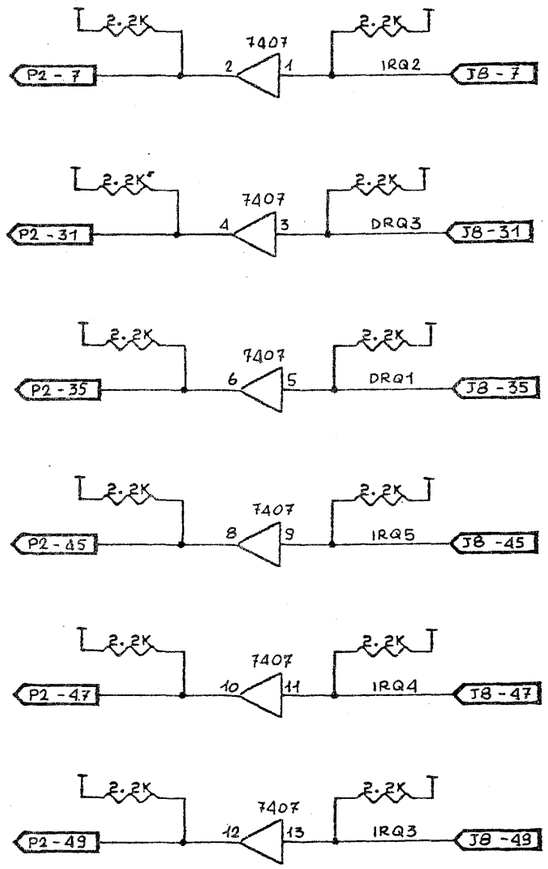
OLIVETTI PERSONAL COMPUTER M24
POWER SUPPLY

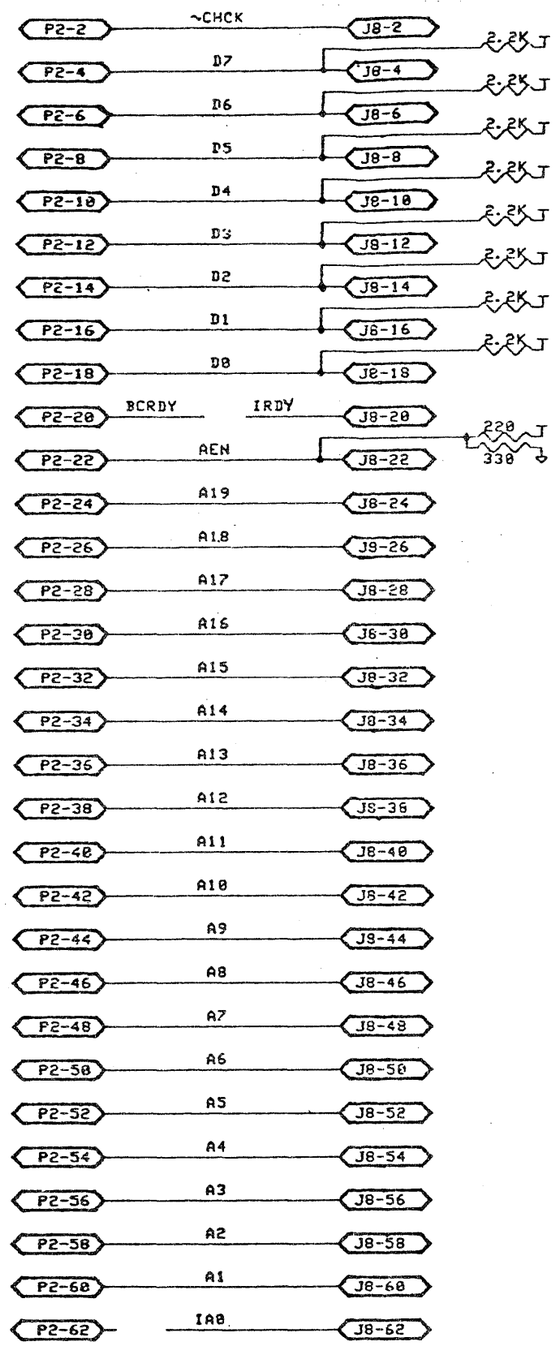


FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
BUS CONVERTER BOARD
PAGE 1 OF 3



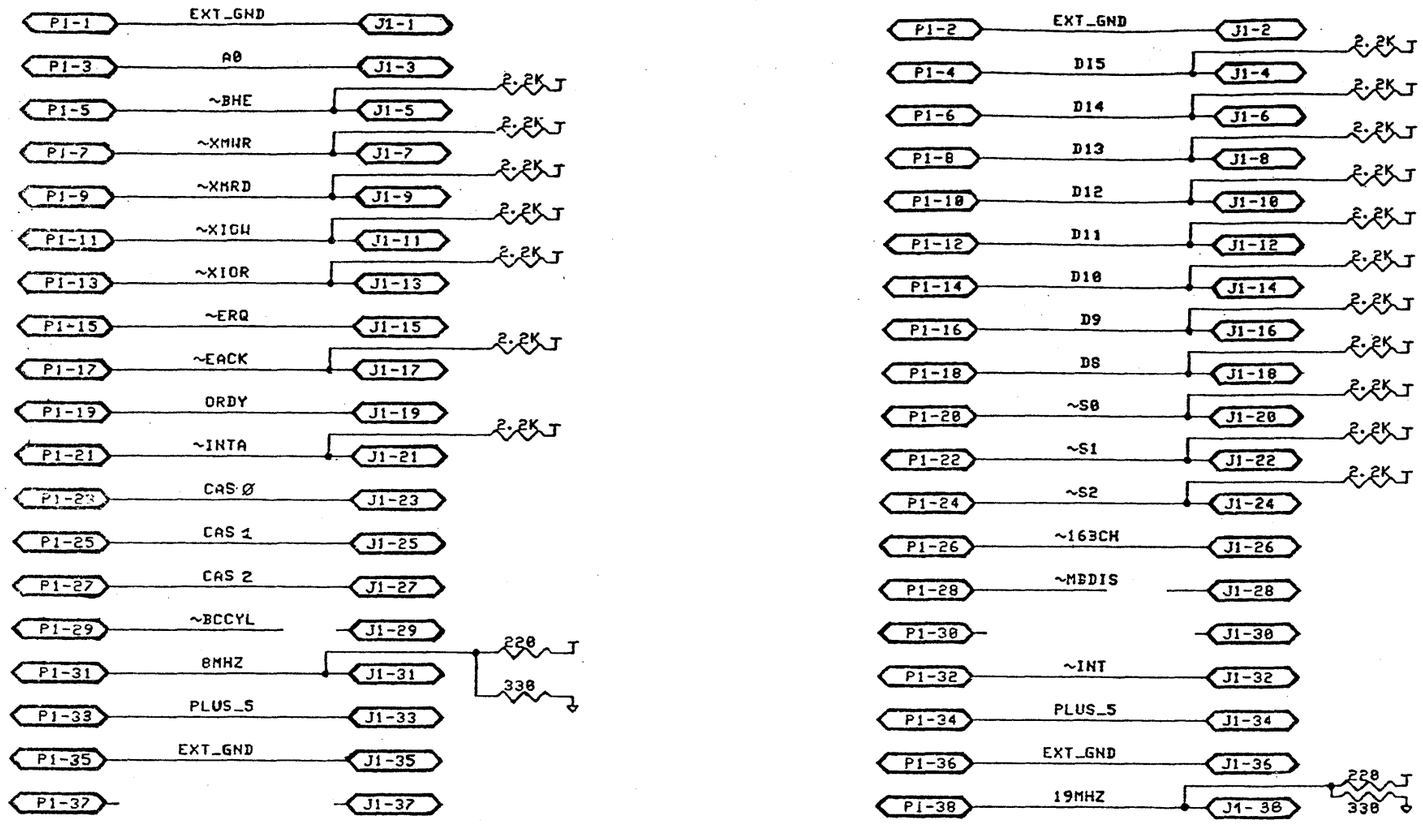




NOTE: J9 THROUGH J14 SAME AS SHOWN FOR J8

OLIVETTI PERSONAL COMPUTER M24
BUS CONVERTER BOARD
PAGE 2 OF 3

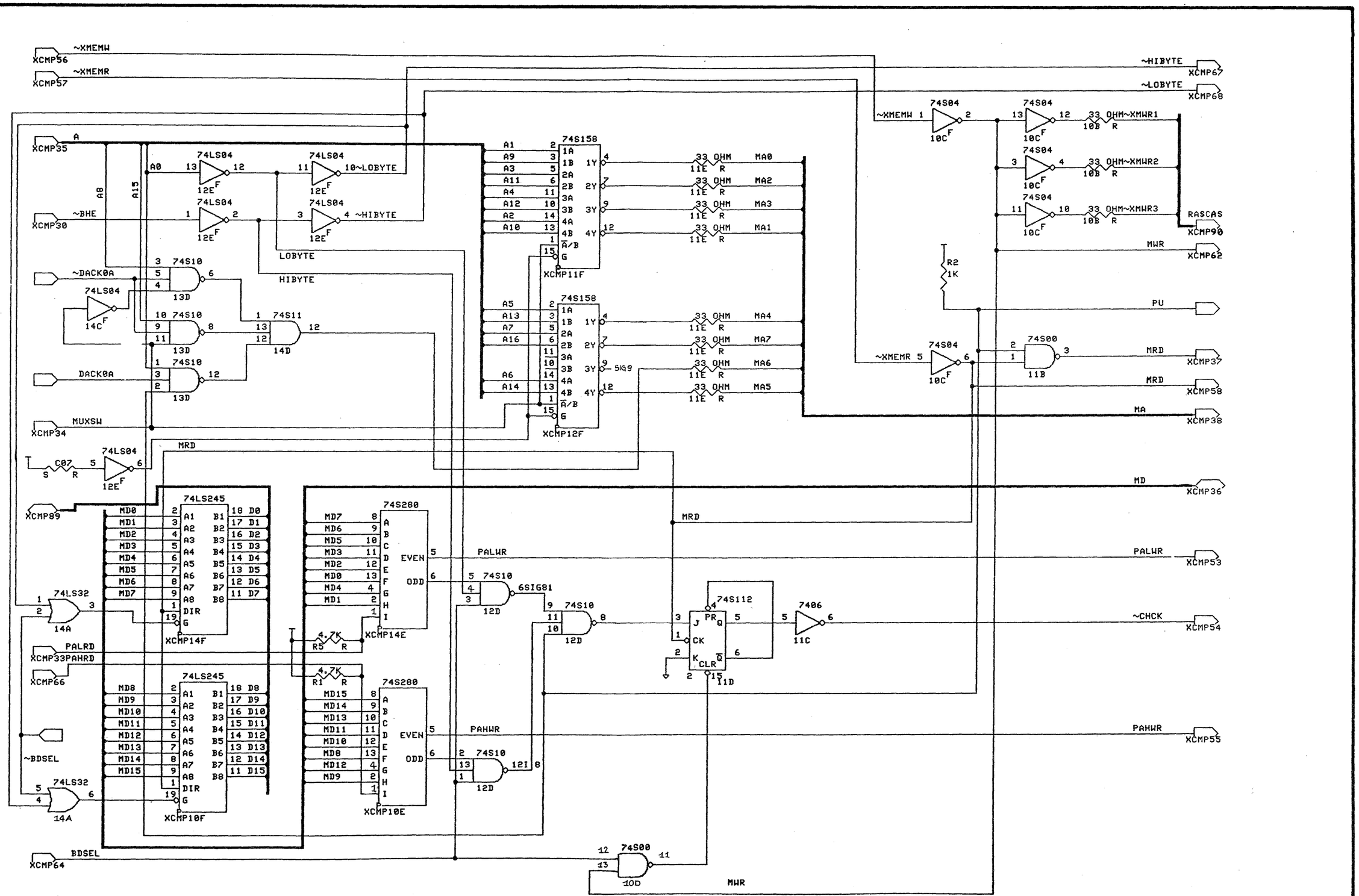
FOR TRAINING PURPOSES ONLY



NOTE: J2 THROUGH J7 SAME AS SHOWN FOR J1

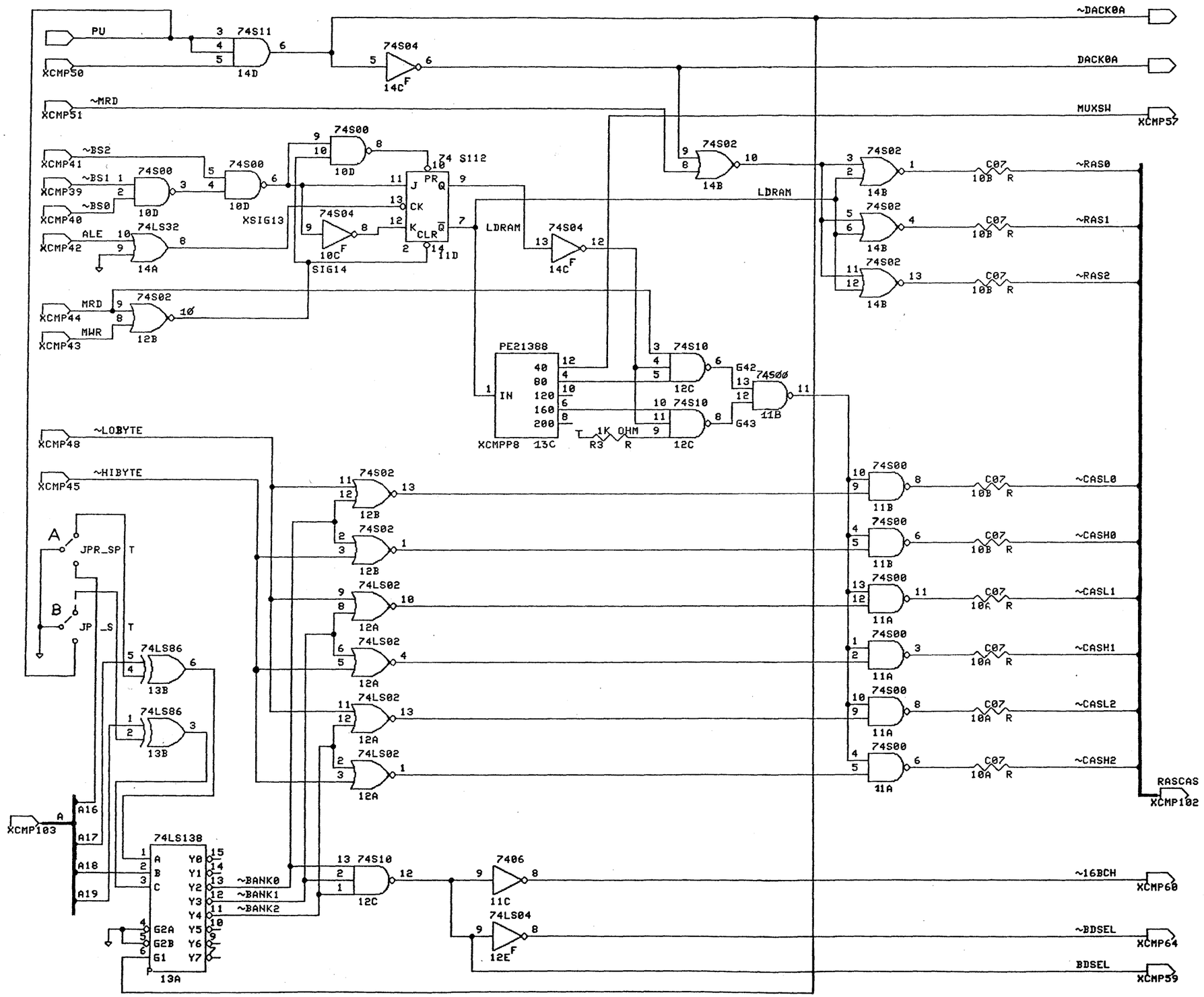
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
BUS CONVERTER BOARD
PAGE 3 OF 3



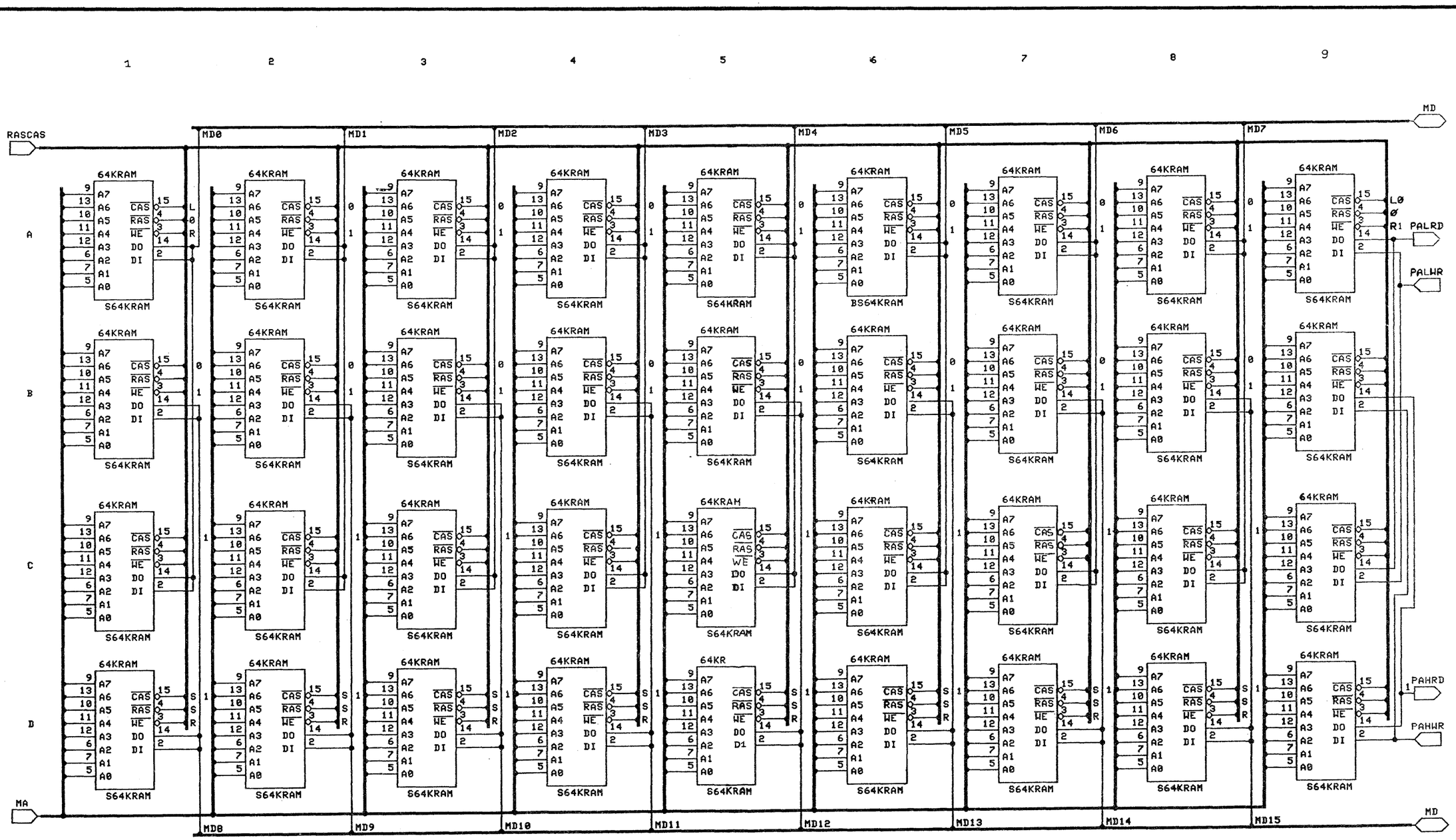
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
MEMORY EXPANSION BOARD



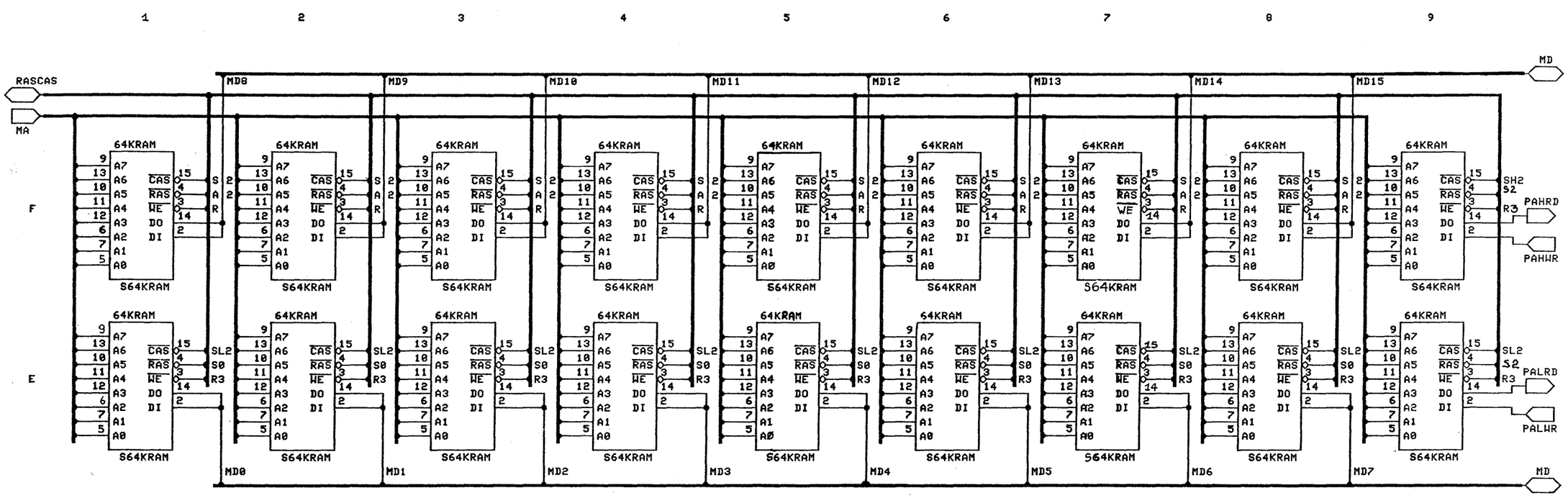
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
MEMORY EXPANSION BOARD
PAGE 2 OF 4



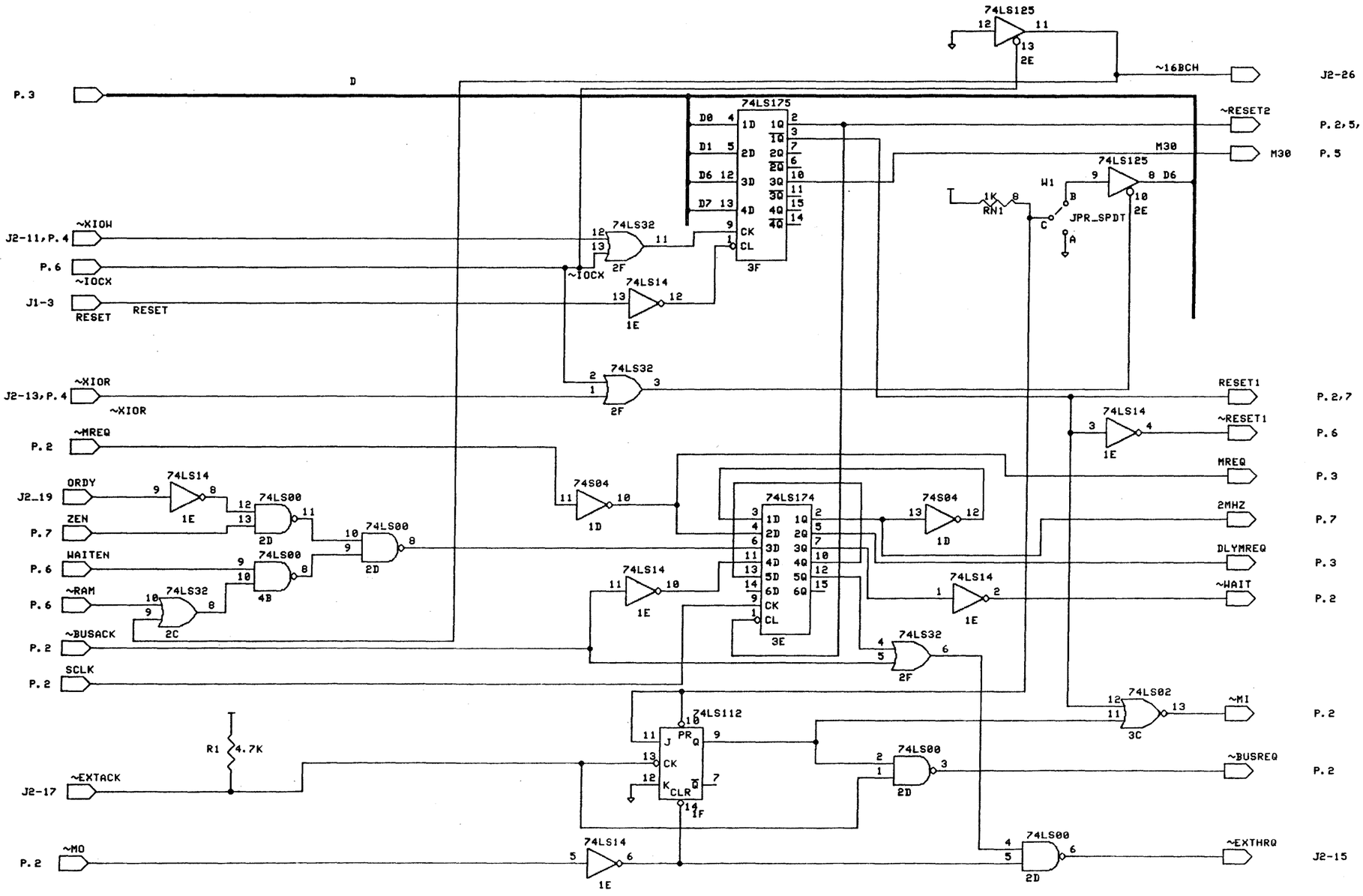
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
MEMORY EXPANSION BOARD
PAGE 3 OF 4



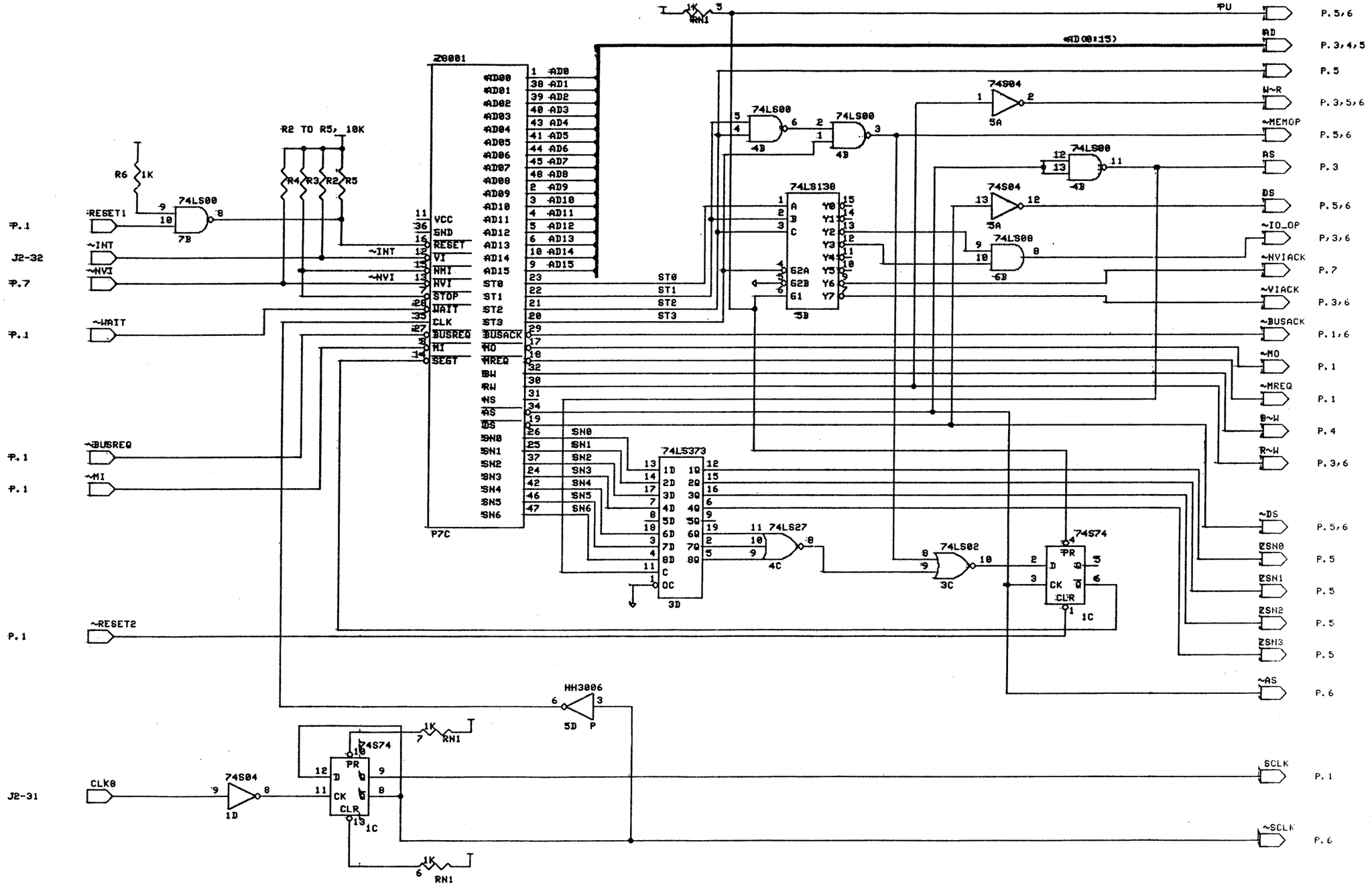
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
MEMORY EXPANSION BOARD
PAGE 4 OF 4



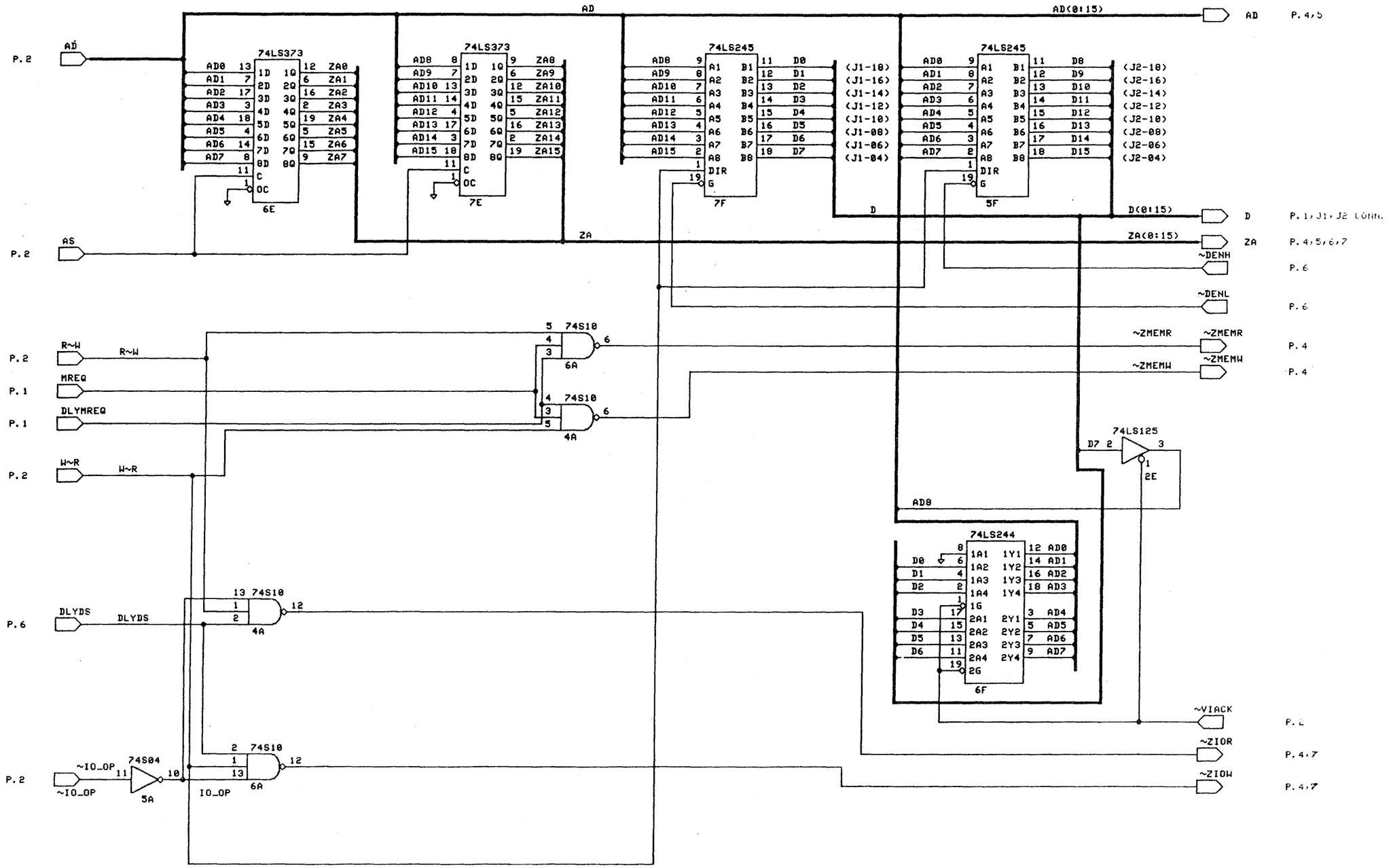
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
APB Z8000 BOARD
REV. P5 PAGE 1 OF 7



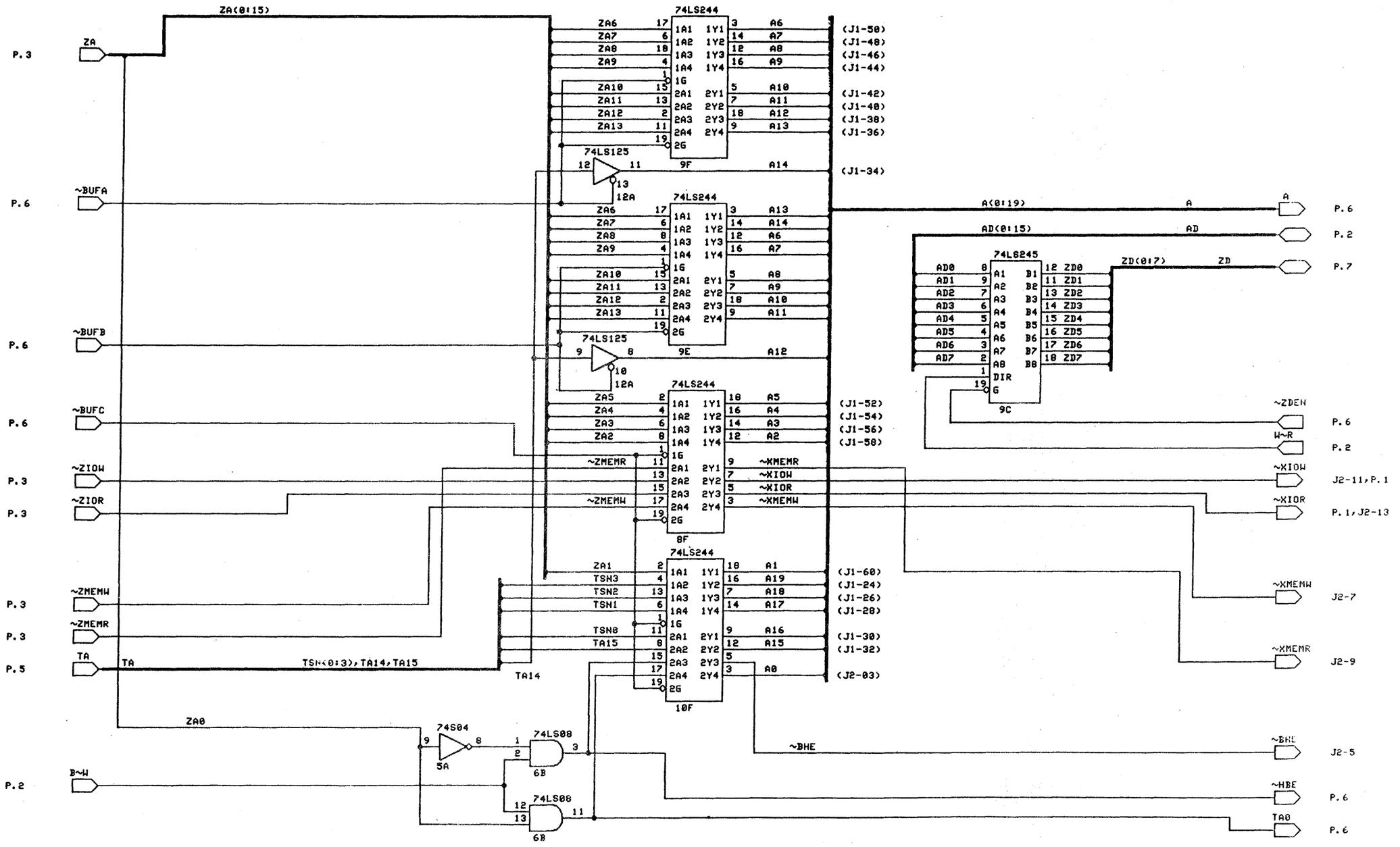
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
APB Z8000 BOARD
REV. P5 PAGE 2 OF 7



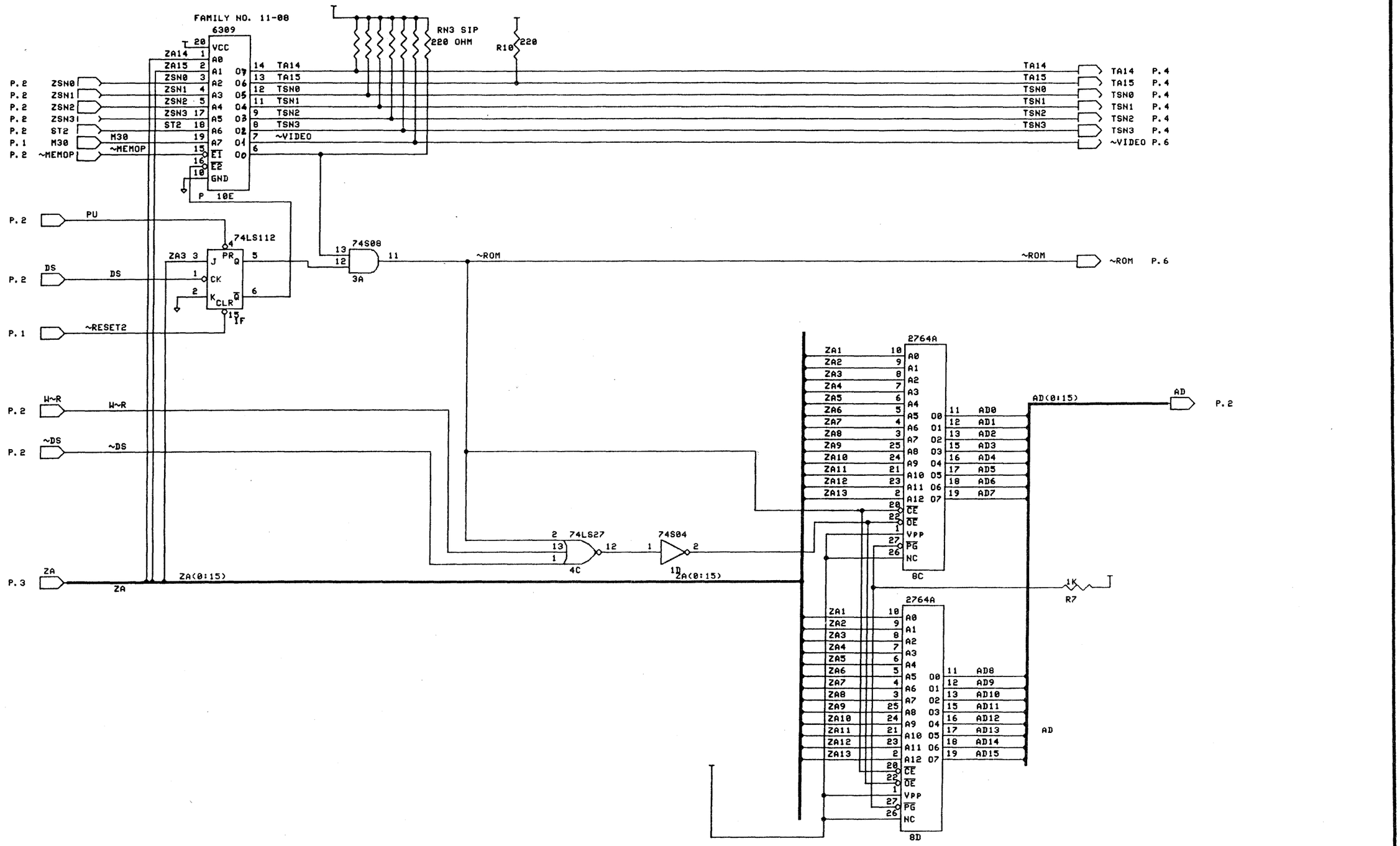
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
APB Z8000 BOARD
REV. P5 PAGE 3 OF 7



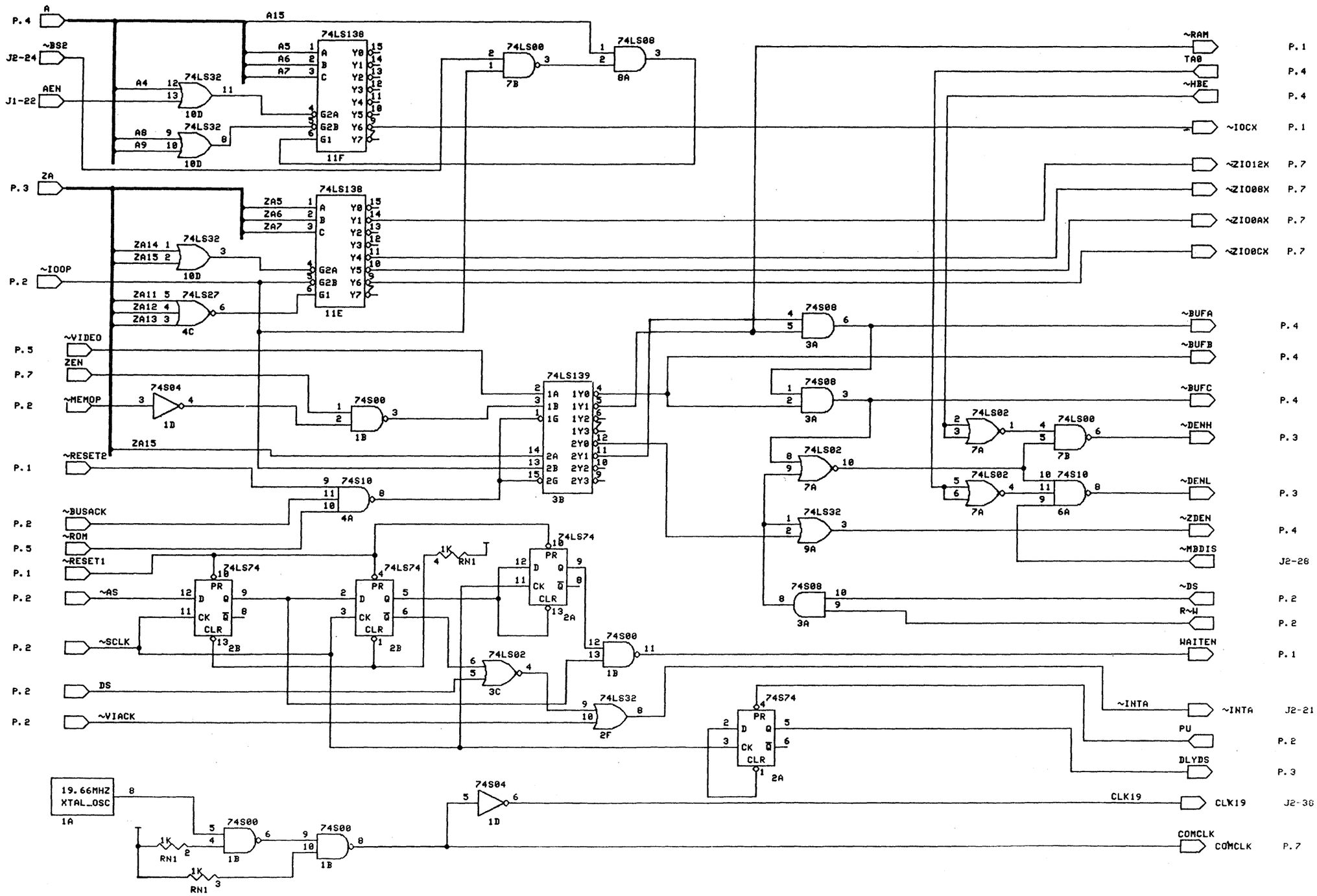
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
APB Z8000 BOARD
REV. B5 PAGE 4 OF 7



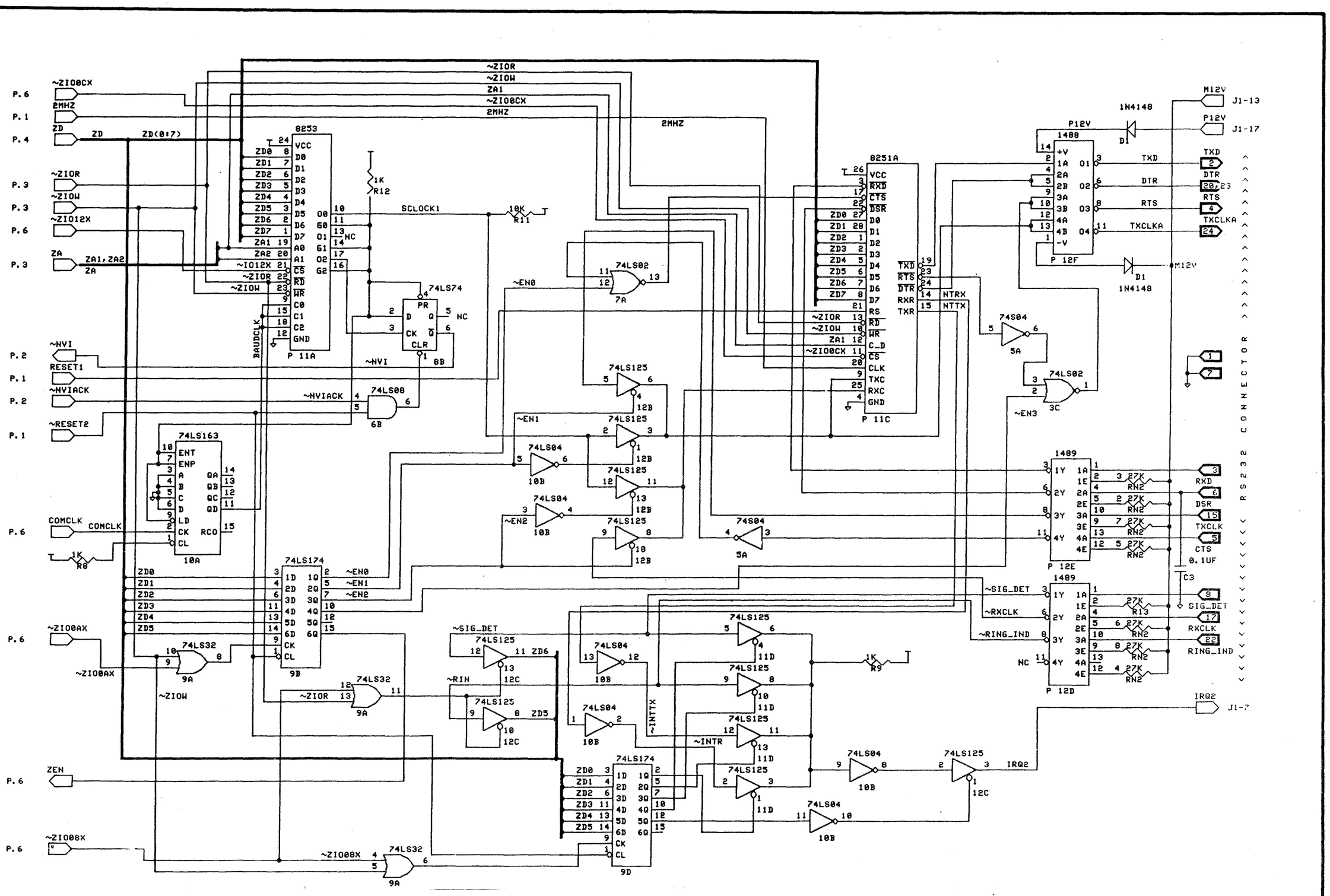
FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
APB Z8000 BOARD
REV. P5
PAGE 5 OF 7



FOR TRAINING PURPOSES ONLY

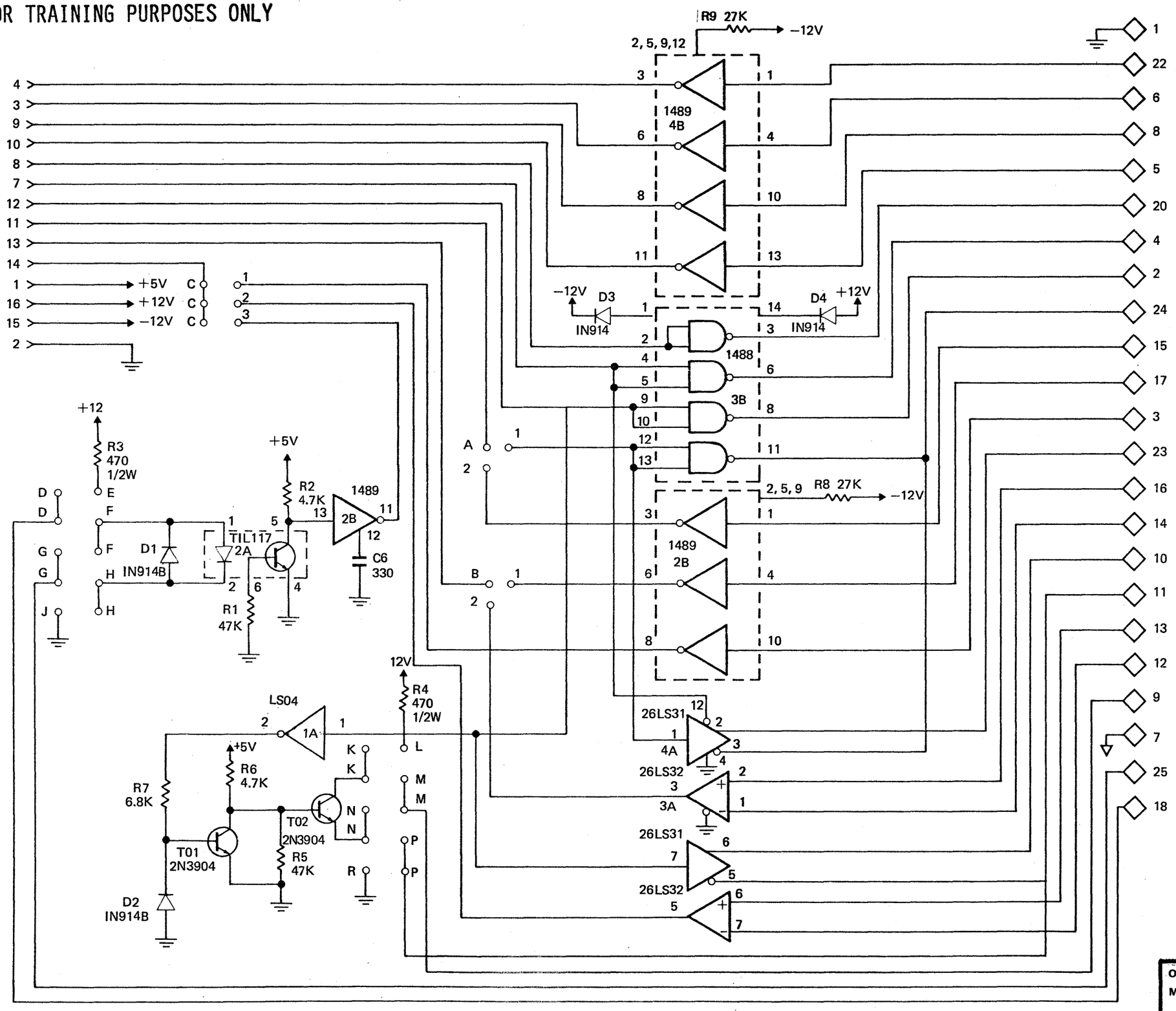
OLIVETTI PERSONAL COMPUTER M24
APB Z8000 BOARD
REV. P5 PAGE 6 OF 7



FOR TRAINING PURPOSES ONLY

OLIVETTI PERSONAL COMPUTER M24
APB Z8000 BOARD
REV. P5 PAGE 7 OF 7

FOR TRAINING PURPOSES ONLY



OLIVETTI PERSONAL COMPUTER M24
MULTI. COMMUNICATION INTERFACE
PAGE 1 OF 1

UPDATING STATUS

DATE	UPDATED PAGES	PAGES	CODE
28-02-84	1st EDITION	234	4100690 C (0)
15-11-84	2nd Edition General Revision, Updating to cover the M21 and addition of Chapter 8 containing information about the APB Z8000 Board	321	4100690 C (1)

Pages marked * must be suppressed

Code 4100690 C (1)
Printed in Italy

olivetti