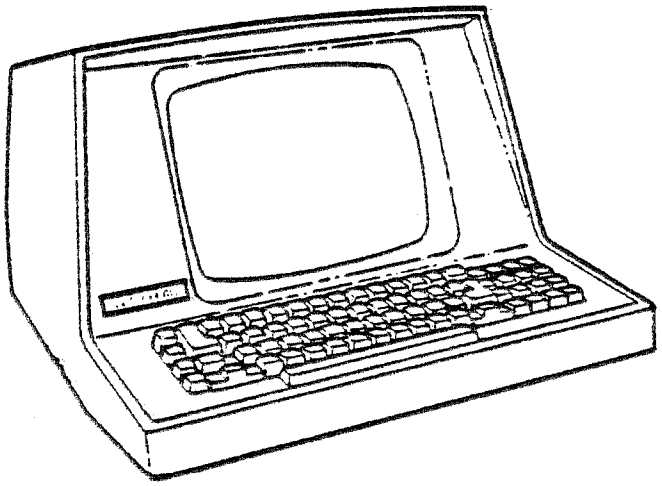
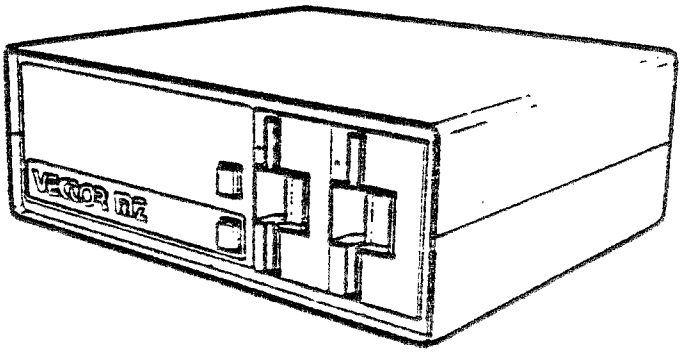
ZCB

Z-80 SINGLE BOARD COMPUTER





VECTOR
VECTOR GRAPHIC INC.

ZCB SINGLE BOARD COMPUTER

Revision 1

USER'S MANUAL

Revision B

June 11, 1980

Copyright 1980 Vector Graphic Inc.

Copyright 1980 by Vector Graphic Inc.
All rights reserved.

Disclaimer

Vector Graphic makes no representations or warranties with respect to the contents of this manual itself, whether or not the product it describes is covered by a warranty or repair agreement. Further, Vector Graphic reserves the right to revise this publication and to make changes from time to time in the content hereof without obligation of Vector Graphic to notify any person of such revision or changes, except when an agreement to the contrary exists.

Revisions

The date and revision of each page herein appears at the bottom of each page. The revision letter such as A or B changes if the MANUAL has been improved but the PRODUCT itself has not been significantly modified. The date and revision on the Title Page corresponds to that of the page most recently revised. When the product itself is modified significantly, the product will get a new revision number, as shown on the manual's title page, and the manual will revert to revision A, as if it were treating a brand new product. EACH MANUAL SHOULD ONLY BE USED WITH THE PRODUCT IDENTIFIED ON THE TITLE PAGE.

Rev. 1-B 6/11/80

REPAIR AGREEMENT

The ZCB Single Board Computer sold hereunder is sold "as is", with all faults and without any warranty, either expressed or implied, including any implied warranty of fitness for intended use or merchantability. However, the above notwithstanding, VECTOR GRAPHIC, INC., will, for a period of ninety (90) days following delivery to customer, repair or replace any ZCB Single Board Computer that is found to contain defects in materials or workmanship, provided:

1. Such defect in material or workmanship existed at the time the ZCB Single Board Computer left the VECTOR GRAPHIC, INC., factory;
2. VECTOR GRAPHIC, INC., is given notice of the precise defect claimed within ten (10) days after its discovery;
3. The ZCB Single Board Computer is promptly returned to VECTOR GRAPHIC, INC., at customer's expense, for examination by VECTOR GRAPHIC, INC., to confirm the alleged defect, and for subsequent repair or replacement if found to be in order.

Repair, replacement or correction of any defects in material or workmanship which are discovered after expiration of the period set forth above will be performed by VECTOR GRAPHIC, INC., at Buyer's expense, provided the ZCB Single Board Computer is returned, also at Buyer's expense, to VECTOR GRAPHIC, INC., for such repair, replacement or correction. In performing any repair, replacement or correction after expiration of the period set forth above, Buyer will be charged in addition to the cost of parts the then-current VECTOR GRAPHIC, INC., repair rate. At the present time the applicable rate is $35.00 for the first hour, and $18.00 per hour for every hour of work required thereafter. Prior to commencing any repair, replacement or correction of defects in material or workmanship discovered after expiration of the period for no-cost-to-Buyer repairs, VECTOR GRAPHIC, INC., will submit to Buyer a written estimate of the expected charges, and VECTOR GRAPHIC, INC., will not commence repair until such time as the written estimate of charges has been returned by Buyer to VECTOR GRAPHIC, INC., signed by duly authorized representative authorizing VECTOR GRAPHIC, INC., to commence with the repair work involved. VECTOR GRAPHIC, INC., shall have no obligation to repair, replace or correct any ZCB Single Board Computer until the written estimate has been returned with approval to proceed, and VECTOR GRAPHIC, INC., may at its option also require prepayment of the estimated repair charges prior to commencing work.

Repair Agreement void if the enclosed card is not returned to VECTOR GRAPHIC, INC. within ten (10) days of end consumer purchase.

FOREWORD

Audience	This manual is intended for computer distributors, or others with at least a moderate technical knowledge of small computers.
Scope	It will describe what the Vector Graphic ZCB Single Board Computer does in the context of a computer system, how to use the board both in Vector Graphic and in other S-100 systems, and how the board circuitry works.
Organization	Each section is written at a uniform level of technical depth. "Perspective" describes WHAT the board does and requires only a moderate knowledge of computer design. "User's Guide" describes HOW to make it do things and assumes the same level of knowledge, plus the ability to solder jumpers and flip switches. "Theory of Operation" discusses WHY the board works and assumes a knowledge of digital electronics.

TABLE OF CONTENTS

Section — Page

Table of Contents

Specifications

I. Perspective

1.1 The ZCB as a system....1-1
1.2 The ZCB as part of a system....1-1
1.3 CPU section....1-1
1.4 EPROM/RAM section....1-2
1.5 I/O section....1-2

1.5.1 Serial Ports Generally....1-2
1.5.2 Serial Asynchronous Communications....1-3
1.5.3 Serial Synchronous Communications....1-3
1.5.4 RS-232C Theory....1-4
1.5.5 RS-232C on the ZCB....1-5
1.5.6 Parallel Ports....1-6

II. User's Guide

2.1 Introduction....2-1

2.1.1 Standard Jumpering and what it does....2-1

2.2 CPU Section....2-1

2.2.1 Running the system at 2 or 4 MHz....2-1
2.2.2 MWRITE enable/disable....2-2
2.2.3 Auto WAIT state enable/disable....2-2

2.3 EPROM/RAM Section....2-2

2.3.1 2708 EPROM Addressing....2-3
2.3.2 Auxiliary Memory Disable....2-4
2.3.3 2716 EPROM Addressing....2-5
2.3.4 2732 EPROM Addressing....2-6
2.3.5 Enable on-board EPROM on boot....2-6
2.3.6 Phantom enable/disable....2-7
2.3.7 Jumper areas K, L and M....2-7

2.4 I/O Section....2-7

2.4.1 I/O Port Addressing....2-7
2.4.2 Address Mirroring disable/enable....2-9
2.4.3 Asynchronous Serial Baud Rate Selection....2-11
2.4.4 How to connect most serial terminals and printers...2-11

Section | Page

2.4.5 How to connect most low speed acoustic couplers.....2-15
2.4.6 Connecting additional RS-232C handshaking lines.....2-15
2.4.7 Using the Parallel Ports.....2-18
2.4.8 Connecting Sprint 3 to ZCB.....2-19
2.4.9 Connecting Vector MP to ZCB.....2-19

2.9 Spare Chip and Patch areas.....2-20

III. Theory of Operation

3.1 System Operation Block Diagram.....3-1
3.2 Serial Ports.....3-2

IV. Schematics

SPECIFICATIONS-System

Compatibility:	Most S-100 systems.
Power Requirements	+8VDC @ 970 ma. (typ.) +16 VDC @ 120 ma. -16 VDC @ 80 ma.
Availability:	Shipped assembled, tested, burned in; no kits.

SPECIFICATIONS-EPROM/RAM

Memory	65536 bytes addressable, 1024 bytes RAM on board, 3 EPROM sockets on board, up to 12K addressable
EPROMs included with board	none
Memory Speed	RAM: 300ns. EPROM: User selected (450 ns. typical)
Memory Types	RAM: 2114 static EPROM: 2708, 2716, 2732
Standard Location of Systems Monitor EPROM	E000H-E7FFH
Power-On/Reset Jump Options	Auto boot on power on/reset, jumps to memory location E000H. Shipped enabled.

SPECIFICATIONS-CPU

Processor	Z-80A
Number of Data Bits	8
Number of Address Bits	16
Instructions	158, including all 78 8080 instructions
Clock speed	2 or 4 MHz, jumper selectable, enabled for 4 MHz.
Interrupts	Z-80 Mode 0 (8080 mode), MODE 1, MODE 2
I/O devices	256 I/O addresses
Dynamic RAM	Supports dynamic memory by sending Z-80 RFSH on bus line 66; fast reset/power on clear signal generated on board.

Static RAM	Fully compatible
MWRITE	Jumper option to generate MWRITE on board Standard: option enabled.
Wait state generation for memories slower than 300 ns.	3 options: generate one wait state on each bus cycle, generate one wait state after each M1 instruction, generate no wait states. Standard: generate one wait state after each M1 instruction.
Bus Load	1 standard TTL load on all inputs
Buffering	Fan out: 15 standard (60 low power shottky)
Phantom:	Output buffer disable compatible with Vector Graphic EPROM/RAM Boards, which generate phantom in response to Power-on-clear (POC). Jumper selectable: on/off. Standard: enabled
Mirroring	Standard: enabled, can be disabled.

SPECIFICATIONS-I/O

Capacity	1 serial RS-232 and 3 8-bit parallel ports programmable as input or output.
Serial port	1, using 8251 controller chip.
Port addresses	Any increment of four from 00H to FFH. Preset addresses are: Data, 04H (echoed on 06H); Control, 05H (echoed on 07H.)
Signal levels	EIA RS-232C
RS-232 handshaking	Typical handshaking is provided, ie. RTS, CTS, DTR, DSR, etc.
Asynchronous	
Rates	110-9600 baud (switch selectable)
Data bits	5 - 8, programmable
Stop bits	1, 1 1/2, or 2, programmable
Parity	Even, odd, or none, programmable

Synchronous	
Rates	DC-56K.
Synch detect	Can be wired for internal or external synch. 8251 SYNDET line is not connected.
Clock	Not now connected to the external world as required for synchronous operation.
Parity	Even, odd, or none, programmable
Data bits	5 - 8, programmable
Sync character	Single or double synch character can be programmed.
Parallel Ports	2-8 bit, 2-4 bit can be programmed as 3-8 bit. Uses an 8255 parallel I/O controller chip.
Port Addresses	Any increment of four from 00H to FFH. Preset addresses are: Port A, 08H; Port B, 09H; Port C, 0AH and the Control Status Register at 0BH.
Latching	Output latched, input not latched.
Signal level	TTL (input = 1 low power TTL load; output drives 1 TTL load).
Number of lines	8 lines per channel, programmable for input or output. +5 VDC and GND are also provided.
Data transfer	Over 100K bytes/second.
Cable	Optional. Has 34-pin female connector and 34-line ribbon cable. No connector is at the other end, allowing user to configure as required. Must be ordered separately.

I. PERSPECTIVE

1.1 The ZCB as a system

The Vector Graphic ZCB Single Board Computer provides the capability of a complete computer system on a single board. The ZCB contains what you would normally find on a CPU board, a PROM/RAM board and an I/O board, all contained on one S-100 bus compatible board.

The ZCB contains a Z-80A microprocessor, 1024 bytes of static RAM memory, sockets for up to 12K of PROM, one serial I/O port and three 8-bit parallel ports. Circuitry is provided to support static or dynamic memories. All S-100 8080 signals are also generated making the ZCB compatible with 8080 based systems. On-board power regulation and filtering is provided using IC regulators and heat sinks for power dissipation. Input loading on the bus is 1 low power TTL load while output fanout is 10 TTL loads.

While the ZCB does replace 3 boards in a typical system, it is not a complete stand alone microcomputer on one board. A terminal device of some sort is needed as well as additional RAM memory, for most applications. The ZCB could, however, be used as a complete microcomputer in many process control applications. In this case, preprogrammed PROMs (up to 12K bytes worth) could be used on a ZCB board that was equipped with a power supply. Sense data could arrive by an input parallel port, be processed by the CPU section of the board and any data output (control signals, for instance) could be communicated to the equipment via the output parallel port. The data input and a summary of actions taken could be communicated to a central data processing location by the serial port. Thus the ZCB might be used as an intellegent node on a complete process control system, which could decide and provide output immediately and provide information to a higher level data processing system.

1.2 The ZCB as part of a system

In most applications, however, the ZCB will be used a part of a complete Vector Graphic microcomputer system. Since 3 boards are replaced by one, significant cost savings are realized. It also reduces the physical size of the overall computer that it is used with by freeing up two S-100 slots. Demands on the power supply are also reduced, thereby lightening the load on the complete system.

The ZCB combines the most used features of 3 boards: CPU board, PROM/RAM board and Serial/Parallel I/O board.

1.3 CPU section

The Vector Graphic ZCB is designed around the powerful Z-80A microprocessor. The board is S-100 bus compatible. All input and output lines are fully buffered. Loading on the bus is no more than one standard TTL load. In addition, the board has the necessary circuitry to work with

dynamic memories, such as Vector Graphic's 64K memory board. Simplicity of design has been stressed to enhance reliability of operation by the use of MSI and LSI integrated circuits.

1.4 PROM/RAM section

The PROM/RAM section features 1K of on-board RAM memory (using 2114's) and up to 12K of PROM, ROM or EPROM. Note: For the purposes of discussion in this manual, the terms PROM, EPROM and ROM may be used interchangably. Their technical distinction makes no difference in how they are used in the ZCB. The PROM may be implemented using either 2708's, 2716's or 2732's. This allows the user maximum flexibility with regard to choice of operating systems, languages, etc. Several jumper options such as jump on power-on/reset are available. Advanced address decoding logic permits a wide range of addressing options.

1.5 I/O section

The I/O section of the ZCB single board computer offers three 8-bit parallel ports or ports programmable as input or output and one RS-232C level serial port for interfacing with multiple perpheral devices such as printers, modems and terminals.

The CPU sends data to the serial and parallel ports via I/O addresses. The I/O addresses used by the board can be changed as a group. The board uses eight I/O addresses from 00H to FFH. You specify the value of the addresses by jumper(s), as listed in Section 2.4.1. Thus, the entire range of 256 possible port addresses may be accessed with the advanced decoding logic built into the Vector Graphic ZCB.

The ZCB Board does not come with any software for specific applications, though Vector Graphic supplies certain items of software that make use of its features. The Vector Graphic Extended Systems Monitor 4.0 will work with the board as it comes from the factory without any modification.

1.5.1 Serial ports generally

Discussion of the serial port centers around the industry standard 8251 USART (Universal Synchronous/Asynchronous Receiver/Transmitter) chip. Much of the flexibility of the ZCB board derives from the flexibility of this chip and its parallel counterpart, the 8255, both of which can be modified via software. You, through software, can control the rate of serial transmission, and the format of the transmitted data. Data is transmitted as ASCII code having between 5 and 8 bits per character, with an optional added-on parity bit (choice of even or odd), and with one start bit and a choice of one, one and a half, or two stop bits per character. Further, using the 8251, the board can handle either asynchronous or synchronous communication. Baud rates, format, handshaking, and whether communication is asynchronous or synchronous, is specified through software and, in some cases, hardware modifications.

It is not within the scope of this manual to detail the functioning of the 8251 USART chip, nor to teach the theory of serial communication. In order to write your own communications software or to modify the ZCB serial port, you will need to be thoroughly familiar with this chip. You can refer to Intel's Application Note #16, entitled "Using The 8251 Universal Synchronous/Asynchronous Receiver/Transmitter," which Intel will provide. This Note is also an excellent reference on basic communications theory. More readily available references on the 8251, but ones that have less to say about communication theory, are the "INTEL 8080 Microcomputer Systems User's Manual," available either from Intel or most computer retail stores, and Adam Osborne's "An Introduction to Microcomputers, Volume II - Some Real Products," also available in many computer stores.

1.5.2 Serial asynchronous communication

You can select the rate of transmission and reception from a choice of 110, 150, 300, 600, 1200, 2400, 4800, or 9600 bits/second. You choose the rate using a small DIP-switch on the board.

If you are writing custom software, there is also a simple way to divide the chosen rate of a port by 4, allowing some software control of the rate without physically opening up the computer. For example, if you have a modem that is switch selectable for either 1200 bits/second or 300 bits/second, you can write a program that enables the operator using the keyboard to change the computer's rate of communication to match that of the modem at any given time.

You can select via software the number of data bits in each ASCII character, selecting either 5, 6, 7, or 8. You can also select the number of stop bits in each character, selecting either 1, 1-1/2, or 2. Finally, you can select whether or not a parity bit is included for each character, and if chosen, whether or not it is even or odd parity. For how to do this in software, see the references given earlier for the 8251 USART.

1.5.3 Serial synchronous communication

You can enable the serial port to communicate in the synchronous mode. Modifications to the board will be required to accomplish this. In brief, the 8251 SYNDET pin and the TxC and RxC clock pins must be connected to the outside world, which is not the case in the standard configuration of the board.

Once set up for synchronous communication, you can select the rate of communication, by using an external clock between 0 and 56K bits/second. As with asynchronous communication, you can select via software the character length, selecting either 5, 6, 7, or 8 bits. You can also select via software whether or not a parity bit is included for each character, and if chosen, whether or not it is even or odd parity. Also via software, you can select separately for each port whether you are using internal or external synchronization, and whether one or two synch characters are used.

1.5.4 RS-232C theory

This manual cannot describe the RS-232C protocol in detail. For a full description, obtain a copy of the RS-232C EIA STANDARD document, published by Electonic Industries Association, Engineering Department, 2001 Eye Street, N.W., Washington, D.C. 20006. Alternately, if you have access to Datapro or Auerbach reports on communications, they contain thorough articles describing the protocol and its implications. The following information, however, will be of immediate relevance in this manual:

An RS-232C signal can either be POSITIVE (+12 Vdc) or NEGATIVE (-12 Vdc). Positive is ON or SPACING, Negative is OFF or MARKING. (These terms are industry wide conventions that date back to the days of key telegraphy.) RS-232C line drivers typically invert these signals when they are converted to and from TTL signals. Hence, RS-232C POSITIVE corresponds to TTL low (about 0 Vdc) and RS-232C NEGATIVE corresponds to TTL high (about 5 Vdc).

An RS-232C cable consists of 25 lines. An RS-232C transmit or receive data line carries a serial sequence of POSITIVE and NEGATIVE pulses that correspond with the characters you want to transmit or receive. There is also associated formating and parity information attached to the information by the communication device such as an 8251. In addition to the transmit and receive data lines, there are ground lines, (lines 1 and 7), and there are handshaking lines that are used by communication, terminal, and computer equipment to inform each other of their status (lines 4, 5, 6, 8, 20, 22, and a few others that are rarely used). The full RS-232C protocol also specifies a set of rarely used "secondary" lines which have the same definitions as some of the primary lines, but carry an independent set of signals. Altogether there are 25 RS-232C lines defined, but most applications use only a few of them.

In the real world, very few devices require "full RS-232C" protocol. In fact, very few devices even require all of the handshaking lines mentioned above. Many require one or even none. Further, many devices use handshaking lines differently than defined by RS-232C, violating the protocol. In short, it is confusing at this time to say that a given device requires "full RS-232C." You must specify exactly what signals it sends and expects to receive on each line.

It is important to understand that most of the RS-232C lines are directional, that is, the protocol specifies which direction the signal travels on each line, relative to the ends of the cable. Therefore, the protocol specifies that at one end of an RS-232C cable there must be a device of the type called "Data Communications Equipment", or "DCE" for short, and at the other end there must be a device of the type "Data Terminal Equipment, or "DTE" for short. The direction of the signal on a given line can be determined once you decide which end of your cable has which kind of device.

The terms Data Communication Equipment and Data Terminal Equipment derive

from the original purpose for RS-232C - to connect a terminal with a communication device such as a modem. A computer does not have to be involved at all. Since a computer can either play the part of a terminal, when connected to a modem, or it can play the part of communication equipment, when connected to a terminal, a computer serial port can be used either as DCE or DTE. However, a given serial port can only be wired up as one or the other at any one time. If the port happens to be wired up to look like DCE, and you want to connect it to another DCE such as a modem, then the RS-232C connection will not work. Both ends would be transmitting on the same lines and receiving on the same lines. Before the RS-232C connection can be made, in this case, you must rewire the computer's serial port so that it receives and transmits on the lines specified for DTE.

1.5.5 RS-232C on the ZCB

To enable the serial port to communicate over an RS-232C line, you will connect one end of this line to the serial port socket on the ZCB board, and the other end, having an RS-232C standard DB-25 female connector, to the back panel of the computer. The cable is designed so that appropriate signals from the board are directed to the RS-232C lines at the DB-25 connector as if it were Data Communications Equipment. Thus, the resulting DB-25 socket at the rear of the computer is a DCE RS-232C port.

Drivers are provided for the serial port to enable the 8251 Transmit Data and Receive Data lines to input or output at RS-232C voltage levels. These receivers and drivers are already connected on the board, requiring no jumpering. When a serial I/O cable is installed these signals are connected to RS-232C lines 3 and 2 respectively.

In addition, four of the RS-232C control lines are connected to the appropriate pins on the 8251. These are DTR, DSR, RTS and CTS. See the table "Additional RS-232C handshaking lines available" in Section 2.4.2 for the pin assignments. When a serial I/O cable is installed, these signals are available on the cable's DB-25. There are four other signals available from the 8251 which can be accessed: TxRDY, TxMT, SYNDET and RxRDY. You can install jumpers and RS-232C line drivers to enable the 8251 to dynamically control any two of them via software.

Further, a number of other RS-232C lines are available on the board using the standard serial I/O cable mentioned above. These are both input and output lines, but they are not connected to anything other than pads on the board, nor are drivers and receivers connected to them. The table "RS-232C and Connections on ZCB Board" in Section 2.4.6 lists the functions of each of these lines.

Any RS-232C line can be connected to +12 VDC on the board. In addition, for each port, the 8251 can be used to monitor in software any one RS-232C handshaking line, and the 8251 can be used to control from software the output of any two RS-232C handshaking lines, and lastly, the 8251's transmitter can be disabled or enabled by any one RS-232C incoming

handshaking line. There is one spare RS-232C receiver and one spare RS-232C driver available on the board which can be used to connect one input and one output handshaking line.

For the large majority of applications, no additional RS-232C lines will be required other than those already connected to active components on the ZCB board. Thus the serial ports can very often be used as DCE RS-232C input/output ports without modification. Most serial printers, and terminals, can be connected with little or no difficulty.

To connect to a modem, acoustic coupler, or other kind of Data Communications Equipment, a serial port must be converted into a DTE RS-232C port. This can either be done by changing the board wiring slightly or by attaching a Null Modem Cable to the external DB-25, which accomplishes the same thing. However, if RS-232C handshaking is required, other than the lines already connected, then additional modifications to the board will be necessary as explained in Section 2.4.5.

Of course, software is necessary in order to operate specific devices connected to the serial port. Section 2.4 gives the standard I/O port addresses and instructs you how to change them as necessary. Other documents from Vector Graphic describe the particular I/O addresses and peripheral devices which each Vector Graphic software product controls.

For example, at the time of this writing, any Extended Systems Monitor having option C enables a standard serial terminal to be plugged directly into the serial. The Version 4 Extended Systems Monitor (any option) contains a program that enables the operator to communicate to a time share service via a modem connected via an RS-232C cable to the serial port which has to be converted to a DTE port first. Lastly, operating systems such as MDOS and CP/M from Vector Graphic, and Vector Graphic's Word Managagment System, output to printers via the serial port.

1.5.6 Parallel Ports

The ZCB has three independent parallel input or output ports, called Parallel Ports A, B and C. They are connected to the outside world via one 34-pin connector which is mounted on the top of the card. Vector Graphic supplies an optional 34-line ribbon cable that connects to this edge connector. The other end of this cable has no connector on it, allowing you to configure it as desired. Many of the 34 lines are not used, as discussed below.

Each parallel port has eight pins which can be programmed for input, output or, in the case of Port C, control. 16 different combinations are possible in Mode 0, more are available in Modes 1 and 2. See Intel's 8080 Microcomputer Systems User's Manual for more information on Modes 1 and 2. Additional pins provide +5 VDC and GND. Exact pin assignments are given in Table 12 in Section 2.8.

Input and output is accomplished using IN(put) and OUT(put) machine

language instructions within software prepared for specific applications. Output is latched on the board, so that after an OUT instruction is executed, the eight bits of data remain available to the external device until the computer changes it. Input is NOT latched, so that software must be written to sense in some way that data is available and to input that data. There are no interrupt lines connected to the parallel ports.

II. USER'S GUIDE

2.1 Introduction

The User's Guide explains how the board functions as manufactured and tells how to change various user selectable options by means of jumpers and switches to fit other than standard requirements. The User's Guide is divided into 3 sections to cover the three main areas of board operation: CPU section, EPROM/RAM section and I/O section.

2.1.1 Standard jumpering and what it does

The Vector Graphic ZCB single board computer has been prejumpered at the factory to fit current Vector Graphic Microcomputer Systems. If you wish to use the ZCB in another model computer or one of your own design, the following information will prove useful.

The standard jumpering does the following:

2708 EPROM's are selected for 2 PROM sockets.
2716 EPROM selected for 1 PROM socket.
EPROM base address is E000.
1K on-board memory is addressed at FC00.
Serial port is addressed at 04(control) and 05(data)
(Same information is duplicated at addresses 06 and 07)
Parallel ports A,B,C and Control Status Register are addressed at 08,09,0A and 0B, respectively.
Clock speed, 4MHz.
Mwrite is enabled
One wait state is inserted on each M1 cycle.
On-board EPROM is enabled to boot on reset.
Phantom (line 67) is enabled.

2.2 CPU SECTION

2.2.1 Running system at 2 or 4 MHz.

Jumper area : A

Connections as manufactured: pad 1 jumpered to pad 2

Function: selects 2 or 4 MHz. operation. The Z-80 CPU is capable of 4 MHz. operation but some peripheral boards are not. The board is shipped for operation at 4 Mhz.

Options: to operate at 2 MHZ, cut the jumper between pad 1 and 2 and install a jumper between pad 1 and 3.

2.2.2 MWRITE enable/disable

Jumper area : D

Connections as manufactured: jumper installed

Function: when connected, the ZCB board will generate the MWRITE signal (S-100 line 68). You will want to disconnect this jumper if there is some other source of MWRITE in the system, such as a front panel.

Options: if MWRITE generation is not wanted, cut the jumper in area D.

2.2.3 Automatic WAIT state disable/enable

Jumper area : N

Connections as manufactured: pads 1 and 2 are jumpered (wait on M1)

Function: When enabled, this options permits the automatic generation of WAIT states according to the following. When pads 1 and 3 of jumper area N are jumpered, no automatic wait states are enabled. When pads 1 and 2 are jumpered, an automatic wait state is inserted only after the M1 machine cycle. If no pads are jumpered, one automatic wait state is inserted on every bus cycle

Options: To insert one wait state after every bus cycle, cut the jumper between pads 1 and 2. To disable wait state generation by the ZCB board cut the jumper between pads 1 and 2 and install a jumper between pads 1 and 3.

2.3 EPROM/RAM SECTION

The Vector Graphic ZCB Single Board Computer presents the user with unprecedented flexibility in EPROM type choice and address selection. You can choose from 2708, 2716 or 2732 type EPROMs, giving the user from 3K to 12K bytes of EPROM on the single board. The EPROM type chosen directly determines the addressing scheme used. See the appropriate section below for the EPROM type you have chosen.

See appendix section for important imformation regarding PROM addressing on current ZCB boards.

2.3.1 2708 EPROM Addressing

2708 type EPROMs are the easiest type of PROM to chose...they are the factory standard and the board is prejumpered to use them. The standard jumpering of the ZCB provides for the Vector Graphic 4.0 Monitor addressed at E000H.

The board is prejumpered to use 2708's at the following memory addresses:

PROM 1	U20	E000H-E3FFH
PROM 2	U21	E400H-E7FFH
PROM 3	U22	EC00H-EFFFH
RAM	U24,25	FC00H-FFFFH

The span between F000H and FBFFH is taken up, in the standard Vector System, by the Flashwriter II board and the Disk Controller board. There is a 1K gap in memory from E800H to EBFFH if standard jumpering is used.

If you want to change EPROM/RAM memory addresses within the memory address block E000H to FFFFH, use the following procedure.

1) Subtract E000H from the desired PROM or RAM base address.

2) Consult the 1K column of the Relative Address Chart (below) to determine the jumper area F pad number which corresponds to the result obtained in step 1.

3) Consult the Area F EPROM/RAM Socket Pad Chart to determine the pad number of the particular socket you wish to change.

4) Solder a jumper between the two pads.

5) Cut away any pre-existing jumpers as necessary.

Relative Address Chart

Increment	1K	2K	4K
Area F Pad No.	(all addresses in hex)		
14	0000-03FF	0000-07FF	0000-0FFF
13	0400-07FF	0800-0FFF	1000-1FFF
12	0800-0BFF	1000-17FF	2000-2FFF
11	0C00-0FFF	1800-1FFF	3000-3FFF
10	1000-13FF	2000-27FF	4000-4FFF
9	1400-17FF	2800-2FFF	5000-5FFF
8	1800-1BFF	3000-37FF	6000-6FFF
7	1C00-1FFF	3800-3FFF	7000-7FFF

Area F EPROM/RAM Socket Pad Chart

Pad 1 connects EPROM 0 (U20) to its address.
Pad 6 connects EPROM 1 (U21) to its address.
Pad 2 connects EPROM 2 (U22) to its address.
Pad 3 connects RAM (U24,25) to its address.

If you want to use 2708's and change memory locations outside of the standard block (E000H-FFFFH) use the following procedure.

1) Determine which 8K memory block you want to use from the 2708 Base Address Chart below.

2) Rejumper Area H as specified and cut away any pre-existing jumpers as necessary.

3) Use the Relative Address Chart (above) to determine the actual EPROM/RAM pad assignments within Jumper Area F as described in the last section.

4) Solder Area F jumpers as required

5) Cut away any pre-existing jumpers as necessary.

2708 Base Address Chart

Addresses	Area H Jumpers	Area I Jumpers	Area J Jumpers
0000H-1FFFH	1-7,2-5,3-9	No change	No change
2000H-3FFFH	1-6,2-5,3-9	Std: 3-6,4-7,5-8	Std: 2-4,5-7
4000H-5FFFH	1-7,2-4,3-9		
6000H-7FFFH	1-6,2-4,3-9		
8000H-9FFFH	1-7,2-5,3-8		
A000H-BFFFH	1-6,2-5,3-8		
C000H-DFFFH	1-7,2-4,3-8		
E000H-FFFFH	1-6,2-4,3-8*		

* Standard Area H Jumpering

2.3.2 Auxiliary Memory Disable

If you want to disable 1 or 2 1-K increments of main memory (perhaps to use an auxiliary disk controller or video board.) jumper pad 4 and/or 5 of jumper area F to the pad representing the absolute address desired.

This memory disable feature will work with whatever EPROM type is chosen except that the interval disabled will be 2K if the board has been configured for 2716's and 4K if the board has been set up for 2732's.

3.3 2716 EPROM Addressing

To use 2716's in the ZCB, follow the procedure outlined below.

1) Consult the 2716 Base Address Chart (below). This shows the base addresses allowable with 2716's.

2716 Base Address Chart

Addresses	Area H Jumpers	Area I Jumpers	Area J Jumpers
0000H-3FFFH	1-2,2-5,3-9	8-4,7-3,6-2	3-4,6-7
4000H-7FFFH	1-2,2-4,3-9	8-4,7-3,6-2	3-4,6-7
8000H-BFFFH	1-2,2-5,3-8	8-4,7-3,6-2	3-4,6-7
C000H-FFFFH	1-2,2-4,3-8	8-4,7-3,6-2	3-4,6-7

2) With the base address chosen, Jumper area H, I and J as per chart and cut away old jumpers as necessary.

3) Subtract the base address from the absolute address desired to determine the relative address. Consult the 2K column of the Relative Address Chart to find the appropriate Area F pad number.

4) Consult the Area F EPROM/RAM Socket Pad Chart to find the second pad number.

5) Solder jumper(s) as required.

6) Cut away any previous jumpers as necessary.

For example, let us say that we wish to use 3-2716's on our ZCB board and want to use an auxiliary disk controller with our system. We want to address U20 at A000H, U21 at B000H and U22 at B800H. The 1K of on-board RAM will be addressed at 9800H and the disk controller will be at 8800H. Note that there will only be 1K of memory beginning at location 9800H with 1K blank following it.

To implement this change, we would first consult the 2716 Base Address Chart. Since all these addresses lie within the third block we jumper area H as follows: 1-2, 2-5 and 3-8; area I : 8-4, 7-3 and 6-2; and area J: 3-4 and 6-7. All pre-existing jumpers are cut away.

Now that the base address has been established, we consult the relative address chart and make the following connections in jumper area F: 1 to 10, 6 to 8, 2 to 7 and 3 to 11. (Cutting away previous jumpers as necessary.) To enable the disk controller, pad 4 is tied to pad 13.

2.3.4 2732 EPROM Addressing

To use 2732's in the ZCB, follow the procedure outlined below.

1) Consult the 2732 Base Address Chart. This shows the base addresses allowable with 2732's.

2732 Base Address Chart

Addresses	Area H Jumpers	Area I Jumpers	Area J Jumpers
0000H-7FFFH	1-2-3,3-9	8-3,7-2,6-1	3-4,1-7
8000H-FFFFH	1-2-3,3-8	8-3,7-2,6-1	3-4,1-7

2) With the base address chosen, jumper areas H, I and J as per the 2732 chart and cut away the old jumpers.

3) Subtract the base address from the absolute address desired to find the relative address. Consult the 4K column of the Relative Address Chart to find the appropriate Area F pad number.

4) Consult the Area F EPROM/RAM Socket Pad Chart.

5) Solder jumper(s) as required in Area F.

6) Cut away any pre-existing jumpers as necessary.

Note: The 1K of on-board RAM will now occupy 4K of address space. The first 1K will be populated, the remaining 3K, blank.

2.3.5 Enable on board EPROM on boot

Jumper area : B

Connections as manufactured: jumpered

Function: Optionally causes processor to address the first 3 bytes of EPROM on system start up (boot) or reset. Generally, the first 3 bytes of the EPROM are a jump instruction to the fourth byte on the EPROM. The board is shipped with this option enabled.

Options: to disable this option, cut the jumper between both pads.

2.3.6 Phantom enable/disable

Jumper area : C

Connections as manufactured: jumpered.

Function: Allows generation of phantom on S-100 bus line 67 . When enabled, phantom disables other system memory boards. This is useful when you want to jump to a particular EPROM on system power on/reset.

Options: to disable the generation of the phantom signal cut jumper between both pads of jumper area C.

2.3.7 Jumper areas K, L and M

These jumper areas have been included for future expansion. The standard jumpers will accomodate all three types of EPROMs (2708, 2716, 2732) currently in use.

2.4.0 I/O SECTION

.4.1 I/O Port Addressing

The ZCB supports all 256 port addresses possible with the Z-80 CPU. However, the choice of serial and parallel port addresses must be made from predefined groups of 32 addresses each. This is done by changing the jumpers in one or two jumper areas if addresses other than the factory chosen standard are desired.

The port addresses for the one serial and three parallel ports are selected in such a manner as to coordinate with software that is being used. The hardwired jumpers on the I/O section of the ZCB board must match those specified in software. For this reason, the preconfigured port addresses, that is, those addresses already set at the Vector Graphic plant, match the port addresses called for in our standard software.

Four port addresses are reserved for the one serial port. The serial port addresses are preset for data 04H (echoed on 06H) and control 05H (echoed on 7H.) The second pair of data and control ports simply duplicate the information on the first pair.

Four port addresses are also used for the three parallel ports. They are designated Ports A (08H), B (09H) and C (0AH) while the fourth port address designates the control port or Control Status Register (0BH.) They can be configured as three 8-bit input or output ports or two 8-bit and two 4-bit input or output ports.

Setting the addresses for the serial and parallel ports first involves

selecting the port addresses desired. For standard use in Vector Graphic Microcomputer Systems, the default addresses are adequate. For more involved systems, it may be necessary to change port addresses. This is accomplished by merely changing some jumper connections on the ZCB board. The first thing to do is to decide what port addresses you wish to use. There are eight groups of thirty-two port addresses you can choose from. All the port addresses must be chosen from the same group. They are as follows:

I/O Port Address Groups

Group	Port Address(Hex)	JUMPER AREA G
I	00-1F	9-6, 8-4, 7-2
II	20-3F	9-5, 8-4, 7-2
III	40-5F	9-6, 8-3, 7-2
IV	60-7F	9-5, 8-3, 7-2
V	80-9F	9-6, 8-4, 7-1
VI	A0-BF	9-5, 8-4, 7-1
VII	C0-DF	9-6, 8-3, 7-1
VIII	E0-FF	9-5, 8-3, 7-1

Within these groups, port addresses are assigned in increments of four:

Port Address Increments within Groups

AREA	GROUP							
E PAD	I	II	III	IV	V	VI	VII	VIII
NO.	(all addresses expressed in hex)							
3	00	20	40	60	80	A0	C0	E0
4	04	24	44	64	84	A4	C4	E4
5	08	28	48	68	88	A8	C8	E8
6	0C	2C	4C	6C	8C	AC	CC	EC
7	10	30	50	70	90	B0	D0	F0
8	14	34	54	74	94	B4	D4	F4
9	18	38	58	78	98	B8	D8	F8
10	1C	3C	5C	7C	9C	BC	DC	FC

1) To change port addresses, first decide which group you wish to use and jumper area G as per the I/O Port Address Groups Chart. If you wish to chose an address within Group I, jumper area G will not have to be changed unless the jumpers have been cut. If you wish to choose an address from Groups II-VIII, first cut the standard jumpers in area G and rejumper as per chart G.

2) Once the group address has been jumpered, the actual serial and parallel port addresses can be chosen. If you want the serial base address to be X4 and the parallel base address to be X8 then you do not need to change jumper area E unless the jumpers have been previously changed. To set the address at other than the standard, first cut any pre-existing jumpers and jumper as per the Port Address Increments within Groups chart. To set the serial port addresses jumper from Pad 1 to the desired address pad. To set the parallel port base address jumper from pad 2 to desired address pad.

2.4.2 Address mirroring disable/enable

Jumper area : O

Connections as manufactured: function enabled: pad 3 connected to pad 2

Function: emulates 8080 address mirroring: 8-bit I/O port address is mirrored on the high address bus during an I/O instruction.

Options: To disable address mirroring, cut the jumper between pads 3 and 2 and jumper 1 and 2 together. This will place the I/O device address on the lower half of the address bus and the value of the contents of the accumulator on the upper half of the address bus during a direct I/O instruction. During indirect I/O instructions (including block moves), the value of the contents of register C is placed on the lower half of the address bus (device address) while the value of the contents of register B is placed to the upper half of the address bus.

Additional RS-232 handshaking lines available

The following RS-232 data and handshaking are available at the serial I/O connector.

J-2 Pin Number	DB-25 Equivalent	RS-232 Designation	Direction	Name
14	3	RxD	Out	Transmitted Data
15	2	TxD	In	Received Data
7	20	DTR	In	Data Terminal Ready
11	6	DSR	Out	Data Set Ready
13	4	RTS	In	Request to Send
12	5	CTS	Out	Clear to Send
10	7	GND	NA	Ground

Parallel I/O Connector Pinout Chart

1.	N.C.	11.	N.C.	21.	N.C.	31.	PB7
2.	GND	12.	N.C.	22.	PA7	32.	PB6
3.	PC2	13.	PA0	23.	N.C.	33.	N.C.
4.	PC3	14.	N.C.	24.	N.C.	34.	+5V
5.	PC0	15.	PA2	25.	PB1		
6.	PC1	16.	PA1	26.	PB0		
7.	PC5	17.	PA4	27.	PB3		
8.	PC4	18.	PA3	28.	PB2		
9.	PC7	19.	PA6	29.	PB5		
10.	PC6	20.	PA5	30.	PB4		

Serial I/O Connector Pinout Chart

1.	N.C.	5.	N.C.	9.	N.C.	13.	RTS
2.	N.C.	6.	N.C.	10.	GND	14.	RxD
3.	N.C.	7.	DTR	11.	DSR	15.	TxD
4.	N.C.	8.	N.C.	12.	CTS	16.	N.C.

2.4.3 Asynchronous serial baud rate selection

You select the desired baud rate through a combination of hardware switches and software. If you are using the standard operating systems and Extended Systems Monitors for Vector Graphic Systems, however, you need only be concerned with the hardware switches.

The hardware switch is located in the upper left hand corner of the board. It is labeled "Baud Rate Select". The switch contains eight rockers labeled "1" to "8" and also labeled "9600," "4800," "2400," "1200," "600," "300," "150," and "110." The switch is, in addition, labeled "OPEN" on the left side.

To select one of the labeled baud rates, press the desired rocker down on the right side, i.e. on the side opposite the "OPEN" designation. Then press all other rockers down on the left side, i.e. toward the "OPEN" designation. The result must be that one rocker is down toward the right, and all others are down toward the left. Otherwise none of the serial channels will not work at all.

If you are not using the serial channel, switch them to the "OPEN" position.

The labeled baud rates assume that the corresponding 8251 will be initialized for a clock factor of 16. (It creates the baud rate by dividing the clock input by 16.) This initialization is handled in software, as described in the 8251 references given in Perspective. All standard Vector Graphic software, unless otherwise documented, use a clock factor of 16, and therefore the labeled baud rates are correct.

However, custom software can use a clock factor of 64. If this is the case, the actual baud rate will be 1/4 of the baud rate selected on the switch. (The only two asynchronous clock factors allowed by the 8251's are 16 and 64.) Further, it may be desired to create software which allows the operator to choose between two baud rates that differ by a factor of 4. For example, the software may be designed in conjunction with a modem that can accept signals at either 300 or 1200 baud. In this case, you would create the software so that the operator's input determines whether the 8251 is initialized with a clock rate factor of 64 or 16, respectively.

Remember, if you are using standard Vector Graphic software, do not worry about this software option. You simply set the baud rate for a desired serial channel by setting the baud rate select switch on the ZCB II board.

2.4.4 How to connect most RS-232C serial terminals and printers

To connect the serial channel to an external peripheral, plug the male 16-pin connector of the ZCB serial I/O cable into the 16-pin socket of the desired serial channel, on the upper right side of the board. Then, in most cases connect the DB-25 socket at the end of this cable to the peripheral, using a 25-wire flat ribbon cable. Details (and exceptions) are as follows:

Always insert the connector so that the ribbon cable emerges toward the top of the board. You can double check this by checking that the "1" on the connector corresponds with the "1" printed on the board next to the socket. Then install the DB-25 at the other end of the cable into one of the available sockets at the rear of the computer, or wherever convenient.

The board comes with one serial I/O cable.

Many RS-232C serial terminals and printers, particularly those working at 1200 baud or less, require none of the RS-232C handshaking lines. Some require that one of these handshaking lines be held at +12 VDC. In either case, you can plug these peripherals directly into the DB-25 at the end of the serial I/O cable without modifying the board at all if you are using stnadard Vector Graphic software, because the software holds the most common handshaking lines at +12 VDC - lines 5, 6, and 8. Note that the ZCB does not require any hardware handshaking signals from the peripheral in order to operate.

If you are not sure whether the board will work without modification, try it before attempting to add additional handshaking signals. Generally, to make sure that you are connecting all the necessary lines, use a 25-wire ribbon cable between the DB-25 at the end of the serial I/O cable and the DB-25 connected to the peripheral. The ZCB board does not generate undesired inhibiting signals on any of the lines.

NOTE:

If you are writing your own software, you will have to program the 8251 on the ZCB to control any handshaking lines required by the peripheral device. If it is simply a matter of holding high or pulling low a line, you can simply strap the appropriate pin with a source of +12V or GND at the connector before it even comes into the computer. Lines controllable by software are listed in the table: Additional RS-232 handshaking lines available.

If a 25-wire cable is not possible, then attempt the connection using three wires, connecting RS-232C lines 2, 3, and 7 from the DB-25 at the end of the serial I/O cable to the same pins on the DB-25 which plugs into the peripheral. (Receive-only printers that do not generate acknowledgement signals, such as Teletype or Decwriters, do not require connecting line 2. It is never necessary to connect line 1 - protective ground - because the ZCB does not ground it.) If this does not work, then the peripheral may require one of the RS-232C handshaking lines. The two most common RS-232C handshaking lines required by serial peripherals are Clear to Send (line 5) and Data Set Ready (line 6). These two lines are held at +12 VDC by standard software. The manual for your peripheral should specify if any are required. To connect one or more, simply connect a wire between the corresponding desired pin numbers on the DB-25's at both ends of your external cable, in addition to the three lines (2, 3, and 7) given above. Remember that you do not have to worry about this if you simply use a 25-wire ribbon cable.

Many peripherals hold either or both lines 4 and 20 at +12 VDC. This allows an alternate method of providing the peripheral with +12 VDC on lines 5, 6, and/or 8. Simply make the appropriate connection(s) at the back of the peripheral. This can be done by soldering jumpers within the DB-25 connector or external to it. For example, to provide +12 VDC on lines 5 and 6, you can connect pin 4 to 5, and pin 6 to 20. This eliminates the need for the extra wires in your external cable, though it is a less elegant solution.

The following describes the external cabling required by three commonly used peripherals. In each case, we are referring to the serial interface version of the peripheral. In each case, do not forget to set the serial port on the ZCB board to the same baud rate that your are setting the printer.

(1) Hazeltine 1400 Video Display Terminal or equivalent:

Use a 25-wire ribbon cable. Alternately, create a 3-wire cable connecting lines 2, 3, and 7 straight through.

Ignore the instruction in the Hazeltine 1400 Video Display Terminal Reference Manual (May 1978) in Section 5.5 which direct you to cross lines 2 and 3 if connecting directly to a computer. (That instruction assumes you are connecting to a computer channel configured as Data Terminal Equipment.) Also ignore the instruction to connect line 1 and ignore the absence of an instruction to connect line 7.

Other Hazeltine terminals are usually connected in the same way.

(2) Diablo 1610 or 1620 printers, NEC Spinwriter with Diablo protocol, Qume Sprint 5, or DataProducts letter quality printer with Diablo protocol:

Use a 25-wire ribbon cable.

Alternately, you can create a 6-wire cable connecting lines 2, 3, 5, 6, 7, and 8 straight through. However, if you are not using the ASCII acknowledgement signals sent by these printers and you are not using a keyboard built into the printer, you do not have to connect lines 2 and 5. Since most Vector Graphic software makes use of the acknowledgement signals (emulating the Diablo protocol), you should connect these two lines when using a Vector Graphic computer.

With at least the Qume, you can use a 3-wire cable connecting only lines 2, 3, and 7 if you connect pins 5, 6, and 8 to pin 20 at the back of the printer.

Note that some models of these printers have female sockets in their rear, in which case you will have to configure a cable with male connectors at both ends.

(3) TI 810 Receive-Only Printer

1200 baud or less:

Use a 25-wire ribbon cable, with male DB-25 connectors at both ends. (TI 810 has a female connector at its rear, as does the ZCB Serial I/O cable.) On both ends of this male-to-male cable, pin 1 is the upper left-hand pin when looking toward the connector's pins and holding the connector so that the longer row of pins is on top.

Alternately, you can create a 2 wire cable connecting lines 3 and 7 straight through, and using a male DB-25 at both ends, as described above. In addition, you MUST solder a jumper between pins 6, 8, and 9 at the printer end of the cable, in order to enable the printer. (The printer provides +12 VDC on line 9, and requires +12 VDC on lines 6 and 8.)

2400 baud or more:

Since the TI 810 printer can print no faster than 150 characters per second, which corresponds to 1500 baud, transmitting at a rate of 2400 baud is too fast for the printer using the cable connection described above. However, the TI 810 can receive at up to 9600 baud. This is feasible if the computer monitors line 11 coming from the printer. The printer pulls line 11 down to -12 VDC when it cannot accept any more characters, and otherwise holds line 11 at +12 VDC. Line 11 is called "printer busy status." Since incoming line 4 on the ZCB board is connected to RTS, connect RS-232C line 11 to line 4 at the computer connector or the printer connector.

What the above accomplishes is to connect the printer busy status line to the CTS input of the 8251 so that when the printer signals it cannot accept more characters, CTS is pulled up, which disables the 8251 transmit circuit which in turn prevents software from sending characters to the 8251. This assumes the software polls the 8251 TxRDY status bit or is interrupt driven by the TxRDY line. The former is true for all serial printer drivers from Vector Graphic at the time of this writing.

Note:

When the computer is sending a great deal of data to the TI 810 directly from memory without any pauses at all, even 1200 baud is too fast. The result will be occasional loss of a line of output. This will not happen with most business software or program assembly, because there are many pauses for disk access and/or calculation. It may happen however when printing editor or word processor text directly from memory. If you expect to encounter this, install the modification given above under "2400 baud or more" if the printer is used at 1200 baud.

.4.5 How to connect many low speed asynchronous acoustic couplers and modems

This section is applicable to many acoustic couplers and modems which carry out asynchronous communications at rates of 1200 baud or less. It is almost always applicable for asynchronous couplers and modems operating at 300 baud or less. Specifically, it is applicable to modems and couplers which require only three RS-232C lines coming from the computer: Transmit Data (line 2), Receive Data (line 3), and Signal Ground (line 7).

Because the ZCB board is wired for direct connection to a terminal, you cannot simply connect the ZCB serial I/O cable to the modem. This is because both the modem and the ZCB serial port in its normal configuration are Data Communications Equipment (DCE), and therefore both expect to receive data on line 2 and to transmit data on line 3. Another problem is that most modems and couplers have female sockets, and the DB-25 connector at the end of the ZCB serial I/O cable is also a female socket.

One solution: create a three line cable with male DB-25 connectors at both ends. Wire line 7 straight across, and cross lines 2 and 3. In other words, connect pin 2 of one connector to pin 3 of the other, and vica versa. Such a cable will work with any modem or coupler requiring only three lines. Connect one end of the cable to the DB-25 of the ZCB Serial I/O cable and the other end to the modem or coupler.

We strongly suggest that you do not modify the ZCB board itself or the ZCB Serial I/O cable, in order to cross lines 2 and 3. By modifying or adapting the external cabling instead, the computer itself remains standard, and the serial ports can easily be used for connecting to a terminal or other kind of peripheral if ever required.

2.4.6 Connecting additional RS-232C handshaking lines

If you are using a terminal which requires handshaking over and above the +12 VDC supplied on lines 5 and 6, or if you are using a modem or coupler which requires any handshaking at all, that is, requires more than a three line connection (lines 2, 3 and 7), then continue reading this section.

The following table lists all 25 RS-232C lines by name, number, and source, and indicates what subset of these are connected to the ZCB board via the ZCB Serial I/O cable. For this subset, the table specifies each line's pin number on the 16-pin socket connected to the end of the ZCB Serial I/O cable. The table also lists those lines which are connected to components or jumper pads, in the factory configuration of the board, and what they are connected to. "U16-13" means it is connected to pin "13" of U16. "GND" indicates the line is connnected to Ground in the factory configuration.

RS-232C name	Source	RS-232C pin number (at the DB-25)	ZCB- Pin number on 16 pin socket J2	Component connected to on the board	8251 pin name, or other if relevant
Protective Ground	Both	1	16	—	—
Transmitted Data	DTE	2	15	U16,13	RxD
Received Data	DCE	3	14	U15,6	TxD
Request to Send	DTE	4	13	U15,11	RTS
Clear to Send	DCE	5	12	U16,1	CTS
Data Set Ready	DCE	6	11	U16,4	DSR
Signal Ground	Both	7	10	—	GND
Received Line Signal Detector	DCE	8	9	—	—
(Reserved for Data Set Testing)	—	9	—	—	—
(Reserved for Data Set Testing)	—	10	—	—	—
(Unassigned)	—	11	—	—	—
Secondary Received Line Signal Det.	DCE	12	—	—	—
Secondary Clear to Send	DCE	13	—	—	—
Secondary Transmitted Data	DTE	14	1	—	—
Transmitter Signal Element Timing (DTE Source)	DTE	15	2	—	—
Secondary Received Data	DCE	16	3	—	—
Receiver Signal Element Timing (DCE Source)	DCE	17	4	—	—
(Unassigned)	—	18	5	—	—
Secondary Request to Send	DTE	19	6	—	—
Data Terminal Ready	DTE	20	7	U15,8	DTR
Signal Quality Detector	DCE	21	8	—	—
Ring Indicator	DCE	22	—	—	—
Data Signal Rate Detector	Either	23	—	—	—
Transmitter Signal Element Timing (DCE Source)	DCE	24	—	—	—
(Unassigned)	—	25	—	—	—

RS-232C and Connections on ZCB Board

It is important to note here that in the RS-232C protocol, any given line has one name, regardless of your point of view. For example, although a modem, which is a kind of Data Communications Equipment ("DCE"), receives its data on line 2, line 2 is still called Transmitted Data. Notice that the names of the lines are more meaningful if you look at them from the point of view of a terminal at the other end of the line, that is from the point of view of the Data Terminal Equipment ("DTE"). According to RS-232C, Data Terminal Equipment transmits on the Transmitted Data line and receives on the Received Data line.

The ZCB board is wired to behave as if it were Data Communications Equipment. This is reasonable because the board is most commonly connected to printers, CRT's, and other kinds of Data Terminal Equipment. This explains why RS-232C line 2 - Transmitted Data - is connected to the 8251 RxD (Receive Data) line, and why line 3 - Received Data - is connected to the 8251 TxD (Transmit Data) line, in the factory configuration.

If you want to connect the board to a modem, that is, you want the board to behave like Data Terminal Equipment, you have to reverse lines 2 and 3 somewhere. The result would be that RS-232C Transmitted Data is connected to the 8251 TxD pin and RS-232C Received Data is connected to the 8251 RxD pin, which is exactly the way Data Terminal Equipment should behave.

To send handshaking signals: Data Communication Equipment connected to the board may require a +12 VDC on some other line (most often line 4 or line 20). Similarly, a piece of Data Terminal Equipment connected to the board may require +12 VDC on some line other than 5 or 6, though this is rare. You can supply +12 VDC as a constant enabling signal, by connecting the desired RS-232C line(s) to +12 VDC through a pull-up resistor.

Alternately, you can allow your software to control the +12 VDC enabling signal(s), if desired. The 8251 has two pins that can be controlled by software. They are pin 23 (RTS) and pin 24 (DTR). Software controls the status of these pins by outputting a command instruction byte to the 8251, as described in the 8251 references mentioned in the Perspective section of this manual. Note that by sending a binary 1 to one of these status lines, the line is turned ON, which is converted by the RS-232C line driver into +12 VDC.

To receive handshaking signals: Data Communications Equipment connected to the board may send +12 VDC handshaking signals on one or more lines, most often RS-232C lines 5 and/or 6. Similarly, Data Terminal Equipment connected to the board may send +12 VDC on one or more lines, most often RS-232C lines 4 and/or 20. You do not have to monitor these lines, necessarily. If you choose to, you can read them through software control. The hardwired line drivers and receivers are already in place. The 8251 has two pins that can be used to receive handshaking signals: pin 17 (CTS) and pin 22 (DSR). Software can monitor the status of DSR by monitoring the appropriate bit in the 8251 status byte. The RS-232C line receiver causes +12 VDC to make this bit a binary 1 (ON). Software can also monitor the CTS status. Rather, if the input to CTS is OFF, the 8251 will not transmit

anything. Software can tell that the 8251 is ready to transmit by monitoring the TxRDY bit in the status byte, or by being interrupted by the TxRDY 8251 output (pin 15). For information on the 8251 status byte, refer to one of the 8251 references mentioned in the Perspective section of this manual.

Remember that if you are connecting a serial port to a modem or other Data Communications Equipment, and you are not using the Null Modem Cable, you should disconnect RS-232C lines 5 and 6 from the +12 VDC source on the board, unless you are positive it poses no danger to the Data Communications Equipment.

If you ARE using a Null Modem Cable, the +12 VDC signals are routed to RS-232C lines 4, 20, and 19 respectively, on which the Data Communications Equipment CAN receive +12 VDC without harm. Therefore, when using a Null Modem Cable, you do not have to disconnect a line from +12 VDC unless you want to control the status of that line from software.

To control an RS-232C handshaking line from software, it must pass from an 8251 output pin to the input of one of the driver circuits in a 1488 quad line driver. It must then pass from the output of this 1488 driver circuit to the RS-232C line.)

To respond to an incoming RS-232C handshaking line, it must pass to the input of one of the receiver circuits in a 1489 quad line receiver. It must then pass from the output of this 1489 receiver circuit to one of the 8251 input pins.

For reference:

1488 = a quad line driver.
1489 = a quad line receiver.

There is one spare receiver circuit and one spare driver circuit on the board. The RS-232C input to the spare receiver is U16 pin 13 and its output is U16 pin 11. The input to the spare driver is U15 pin 2 and its RS-232C output is U22 pin 3. Wire your jumpers directly to these pins, when required.

2.4.7 Using the parallel ports

As shown in the introductory I/O section, Parallel Port A is I/O address x8 (usually 08), Parallel Port B is I/O address x9 (usually 09), Parallel Port C is xA, (usually 0A) and the address of the Control Status Register is xB (usually 0B). These ports can be used as is, without jumpering. You can interface to the 34-pin connector J1 at the top of the board using the ZCB Parallel I/O cable supplied by Vector Graphic. This cable comes with no connector at the far end, enabling customization.

The J1 34-pin connector is numbered beginning with the front left-hand pin as number 1. The back left-hand pin is number 2. 3 and 4 are the next two pins to the right, and so on, with odd pin numbers in the front and even pin numbers in the back. The connector cable is set up so that the lines are numbered 1 to 34 from left to right. The functional definition of each line is as follows:

Parallel I/O Connector Pinout Chart

1.	N.C.	11.	N.C.	21.	N.C.	31.	PB7
2.	GND	12.	N.C.	22.	PA7	32.	PB6
3.	PC2	13.	PA0	23.	N.C.	33.	N.C.
4.	PC3	14.	N.C.	24.	N.C.	34.	+5V
5.	PC0	15.	PA2	25.	PB1		
6.	PC1	16.	PA1	26.	PB0		
7.	PC5	17.	PA4	27.	PB3		
8.	PC4	18.	PA3	28.	PB2		
9.	PC7	19.	PA6	29.	PB5		
10.	PC6	20.	PA5	30.	PB4		

2.4.8 Connecting the Sprint 3 to the ZCB

To connect the Sprint 3 high speed printer to the ZCB, you must have the ZCB-Sprint 3 interface board (P/N 42-3200) available from Vector Graphic, Inc. To install:

1) Plug interface card into top 34 pin parallel I/O connector on the ZCB so that the bulk of the interface card is towards the rear of the computer.

2) Feed flat Sprint 3 Ribbon Cable out through the rear panel of the computer. There is a notch at the top of the panel for this purpose.

3) Plug other end of Sprint 3 cable into connector at rear of Sprint 3. See the Sprint 3 manual for further details.

2.4.9 Connecting the Vector MP to the ZCB

To connect the Vector MP dot matrix printer to the ZCB, you must have the ZCB-MP interface board (P/N 12-3400) and an I/O I cable (P/N 16-1202) available from Vector Graphic, Inc. To install:

1) Plug the interface card into top 34 pin parallel I/O connector of the ZCB so that the bulk of the interface card is towards the rear of the computer.

2) Plug the 24 pin DIP connector end of the I/O I cable into the socket on the ZCB-MP interface card.

3) Fasten the other end of the cable with the DB-25S connector to the rear panel of the computer with the hardware provided.

4) Label the connector "MP Signal Connector."

2.9 Spare chips and patch areas

Socket U26 is large enough for a 14-pin DIP. None of the holes are connected to anything on the board with the exception of GND and +5V.

There are a number of unused gates on the ZCB board which you may find useful to use.

3 tri-state 8097 buffers are available.
- U35: in-pin 12, out-pin 11, controlled by U28,13.
- U36: in-pin 14, out-pin 13, controlled by CDSB
- U45: in-pin 2, out-pin 3, controlled by GND

1 three input 74LS10 NAND gate
- U9: in-pins 1,2,13, out-pin 12

1 74S04 inverter
- U7: in-pin 3, out-pin 4

2 74LS04 inverters
- U14: in-pin 13, out-pin 12
- U43: in-pin 1, out-pin 2

3 two input 74LS00 NAND gates
- U17: in-pins 10,9, out-pin 8
- U17: in-pins 13,12, out-pin 11
- U42: in-pins 4,5, out-pin 6

1 RS-232 level driver, DS1488
- U15: in-pin 2, out-pin 3

1 RS-232 level receiver, DS1489
- U16: in-pin 13, out-pin 11, control-pin 12

III. THEORY OF OPERATION

3.1 System Operation Block Diagram

Since the ZCB combines the circuitry normally found on three different S-100 boards, its operation is somewhat complex. In order to make the theory of operation a bit easier to follow, we are first including a block diagram of the major circuit areas of the ZCB to get an idea of the overall system operation. This explanation will be followed by an area by area breakdown.

Sheet 0-System Block Diagram

When the system is initially turned on the RESET block (sheet 4) pulls the POC line low to let the system know that a reset condition exists. When the system has already been in an on condition, the PRESET line going low will also cause it to issue the POC signal. On board, a reset condition will cause the processor to go through a reset. It also causes Memory Control to issue a Phantom.

When the MEMORY CONTROL block (sheet 4) is given a RESET it issues the Phantom signal which disables all memory on the S-100 bus. If enabled, the block will cause the first three bytes of the on-board PROM to be addressed. Normally, these first three bytes are a jump to instruction which causes the rest of the monitor program to be run. During normal operation, the memory control block reads the address lines during an address cycle to determine whether any of the on board PROM or RAM is being addressed. The jumpers in this seciton permit the use of 1K, 2K or 4K EPROM.

The MEMORY block (sheet 5) contains the on-board PROM and RAM. It is controlled from memory control, the S-100 system control bus (via the S-100 interface block) or the CPU.

The CLOCKS and CLOCK DRIVERS blocks (sheet 6) generate the timing signals required by the CPU and the system. The clock can supply a 4MHz or a 2MHz clocking signal, jumper selectable. Even though not necessary with the Z-80 microprocessor, a phase 2 clock signal is generated to maintain compatability with S-100 equipment designed to be used with the 8080. The clocks block also supplies the base signals (2MCLK and SERCLK) which are divided down to supply all RS-232 baud rates.

The PROCESSOR block (sheet 1) is the heart or, more properly, the brain of the entire microcomputer system. It regulates all processes, addressing, inputs and outputs. On board all address and data signals are sent over MOS level data and address buses. All off board addressing and data communication is done via the S-100 interface block.

S-100 INTERFACE block and the CONTROL SIGNAL BUFFERS block (sheet 8) consist of a number of gates which take processor signals and convert them to system signals. That is, many of the signals required by the system are not generated directly from the Z-80. The signals required by the system that are not available directly are synthesized by the ZCB board. The

reason why there is such a disparity between the signals required vs the signals available are evident when you consider that the S-100 bus was evolved around a system based on the 8080 microprocessor using static memory. Our system is based upon the faster, more versatile Z-80 using dynamic memory. However, the signals required are easily provided for.

The I/O CONTROL block (sheet 6) consist of decoders and flip flops which do two separate things. The flip-flops divide down the clock frequency to provide the apporpriate PCLK signals to the 8251 serial I/O chip to control the RS-232 baud rate. The decoding circuitry is used to tell when the parallel and serial I/O portions of the board have been selected by the appropriate combination of address line and MREQ signals.

The SERIAL/PARALLEL I/O block (sheet 7) contains the most ladvanced technology of the system. An 8251 chip provides the RS-232 serial I/O and handshaking signals through line drivers and receivers which convert TTL level signals to RS-232 level signals. An 8255 chip provides three 8-bit parallel input or output ports. There is also a control status register. The 8255 is programmable and the output lines can be configured in 16 different combinations. (and that's only in the first mode...there are three.)

The S-100 DATA TRANSCEIVER block (sheet 2) consists of a series of data in and data out tri-state buffers which feed into or are fed from the MOS on-board data bus. They convert the MOS level signals used on the on-board data bus to TTL level signals used on the S-100 bus. The MOS and TTL signals are similar with respect to voltage but the TTL lines have more drive capability.

The S-100 ADDRESS BUFFER block (sheet 3) is similar to the data transceiver block but simpler since they only have to be unidirectional (from the processor onto the S-100 bus) rather than bidirectional. All address lines are buffered by 74LS244 tri-state buffers. In addition, circuitry is provided to mirror the lower eight address bits on the upper eight address bits if I/O address mirroring has been enabled.

3.2 Serial ports

The heart of the serial I/O ports is the 8251 USART consisting of independent receiver and transmitter. The function of the transmitter is to accept eight bits of parallel data from the data bus, and convert this to serial data with a wide range of formats, parity and stop bits.

The speed at which data is output in asynchronous mode is controlled by a selectable clock rate derived from the 2 MHz system clock by counters U8 and U1. The clock rate is normally sixteen times the required baud rate (though this can be changed when the 8251 is initialized by software), so a frequency of 153.6 KHz is required for 9600 baud. This gives a frequency division ratio of 13.02, which is rounded to 13, relative to the 2 MHz clock. U7 forms a divide by 13 stage, and the gated terminal count at U14-8 can be selected by a DIP switch for 9600 baud. The other common baud rates,

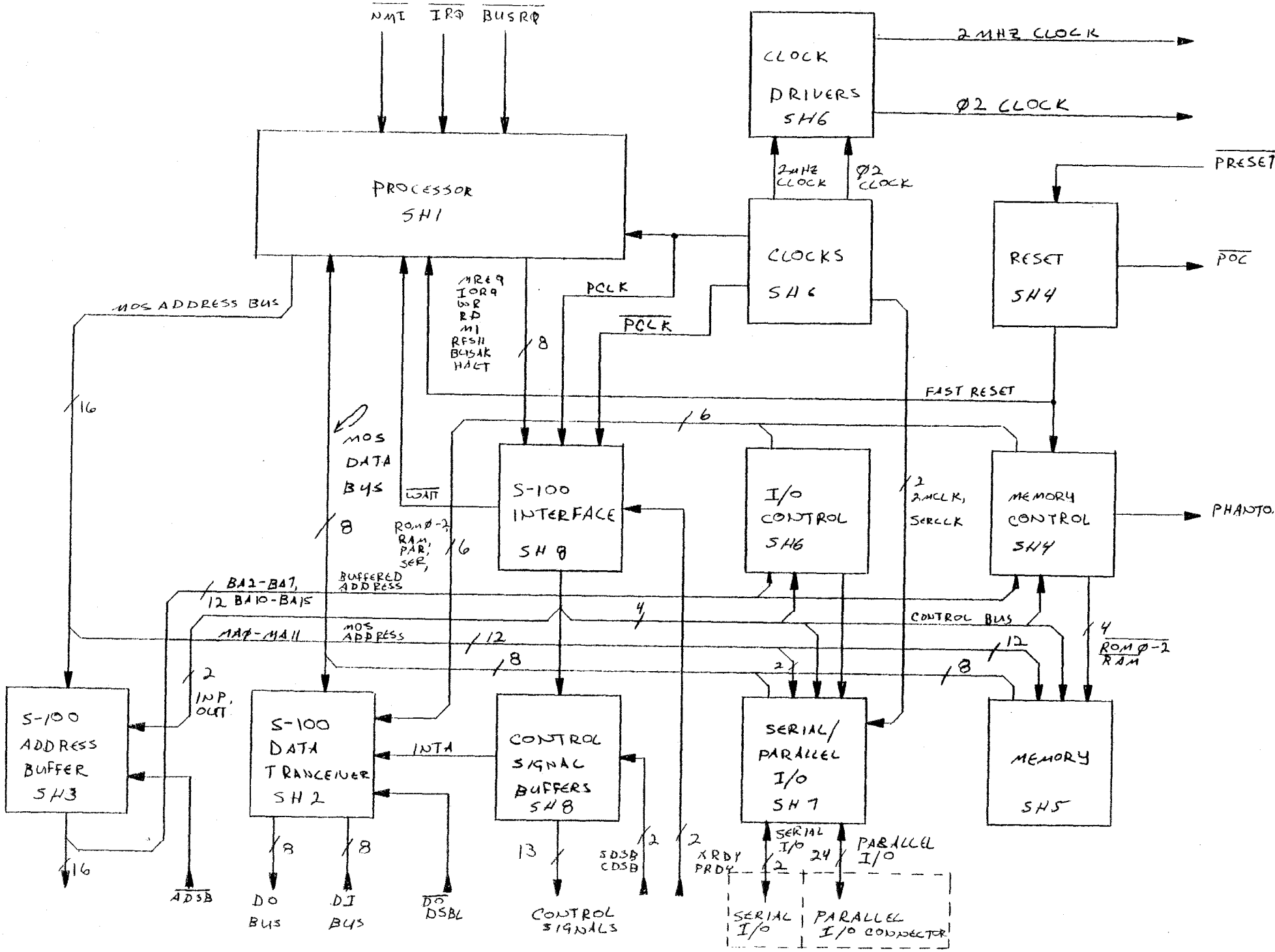
except for 110 baud, are obtained by successively dividing this frequency in half using U1, which is 74LS393 binary counters. The clock frequency for 110 baud is generated by dividing the frequency for 1200 baud by 11. The desired frequency is selected by a DIP switch for each serial port, and applied to the TxC and RxC pins of the 8251.

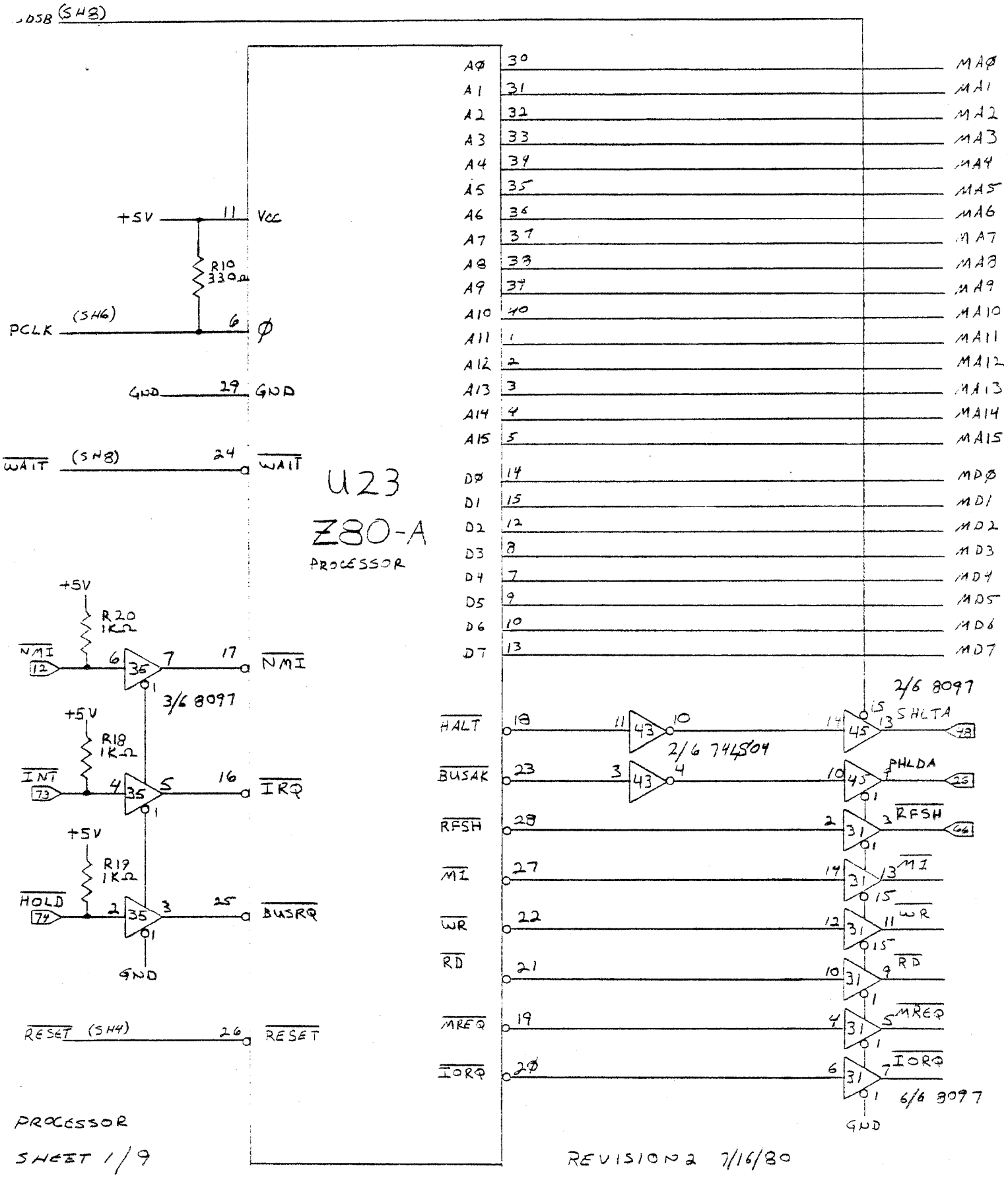
The 8251 USART is designed to interface easily to an 8080 bus structure, and the control signals RD/, C/D, CS/, and WR/ are derived easily from the ZCB internal control bus. Since the device was intended to be used with a bi-directional data bus, EIA RS-232C line drivers and receivers interface the RxD and TxD output of the 8251 to the outside world.

IV. APPENDIX AND SCHEMATICS

APPENDIX

A change has been made in the ZCB PROM socket addressing as the boards are currently being shipped. The third socket (U22) has been reconfigured to accept an 8316E302 type 2K ROM chip addressed at E000H. Sockets U20 and U21 remain configured for 2708's at E800H and EC00H respectively.

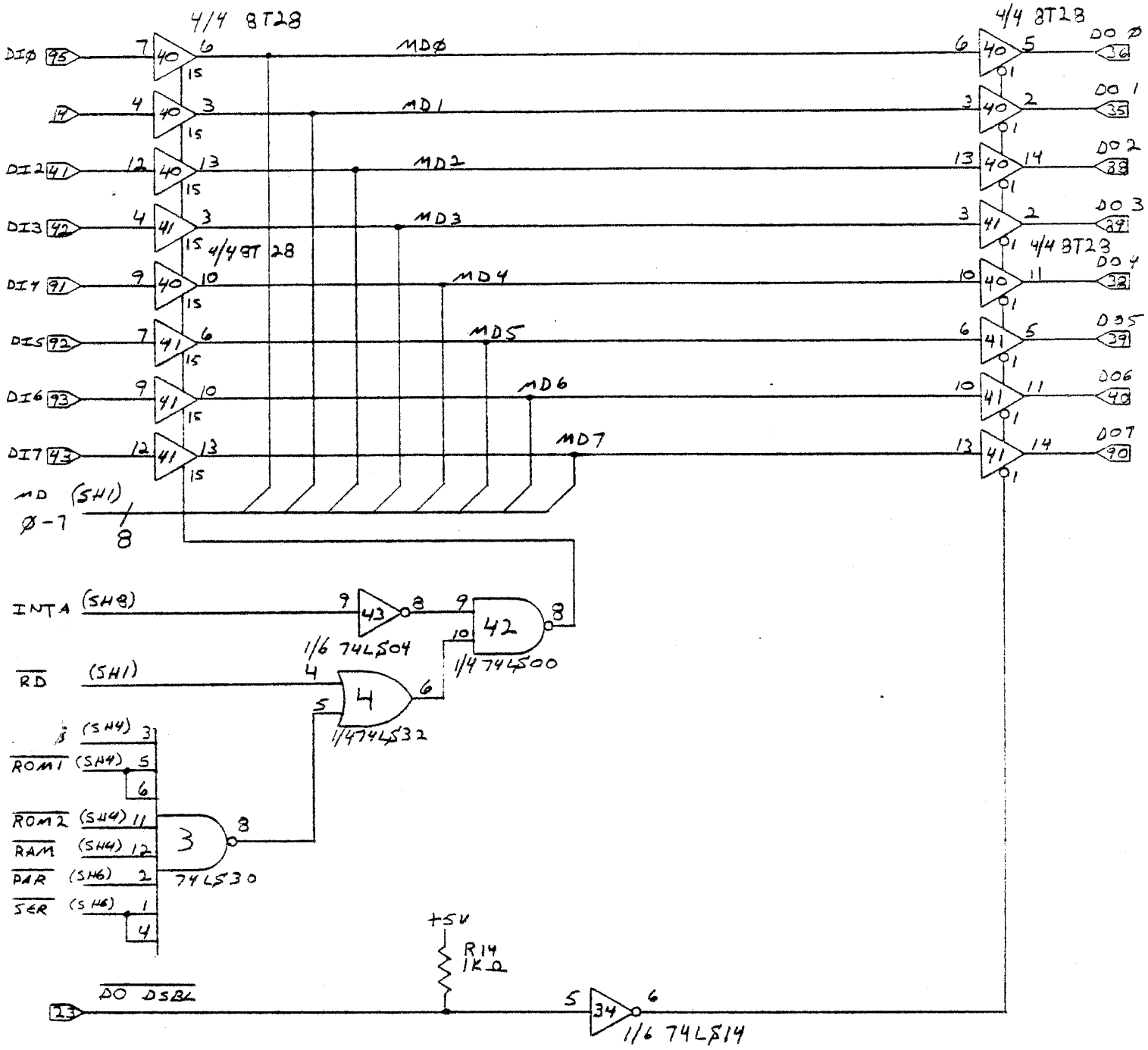

NMI
IRQ
BUSRQ
PROCESSOR SH1
CLOCK DRIVERS SH6
2 MHZ CLOCK
Ø2 CLOCK
2MHZ CLOCK
Ø2 CLOCK
CLOCKS SH6
RESET SH4
PRESET
POC
MREQ
IORQ
WR
RD
M1
RFSH
BUSAK
HALT
PCLK
PCLK
FAST RESET
MOS ADDRESS BUS
MOS DATA BUS
WAIT
S-100 INTERFACE SH8
I/O CONTROL SH6
MEMORY CONTROL SH4
PHANTOM
2 2MCLK, SERCLK
ROMØ-2, RAM, PAR, SER,
BUFFERED ADDRESS
BA2-BA7, BA10-BA15
MAØ-MA11
MOS ADDRESS
CONTROL BUS
ROMØ-2 RAM
INP, OUT
S-100 ADDRESS BUFFER SH3
S-100 DATA TRANCEIVER SH2
INTA
CONTROL SIGNAL BUFFERS SH8
SERIAL/ PARALLEL I/O SH7
MEMORY SH5
ADSB
DO BUS
DI BUS
DO DSBL
CONTROL SIGNALS
SDSB CDSB
XRDY PRDY
SERIAL I/O
PARALLEL I/O
SERIAL I/O
PARALLEL I/O CONNECTOR



PROCESSOR

SHEET 1/9

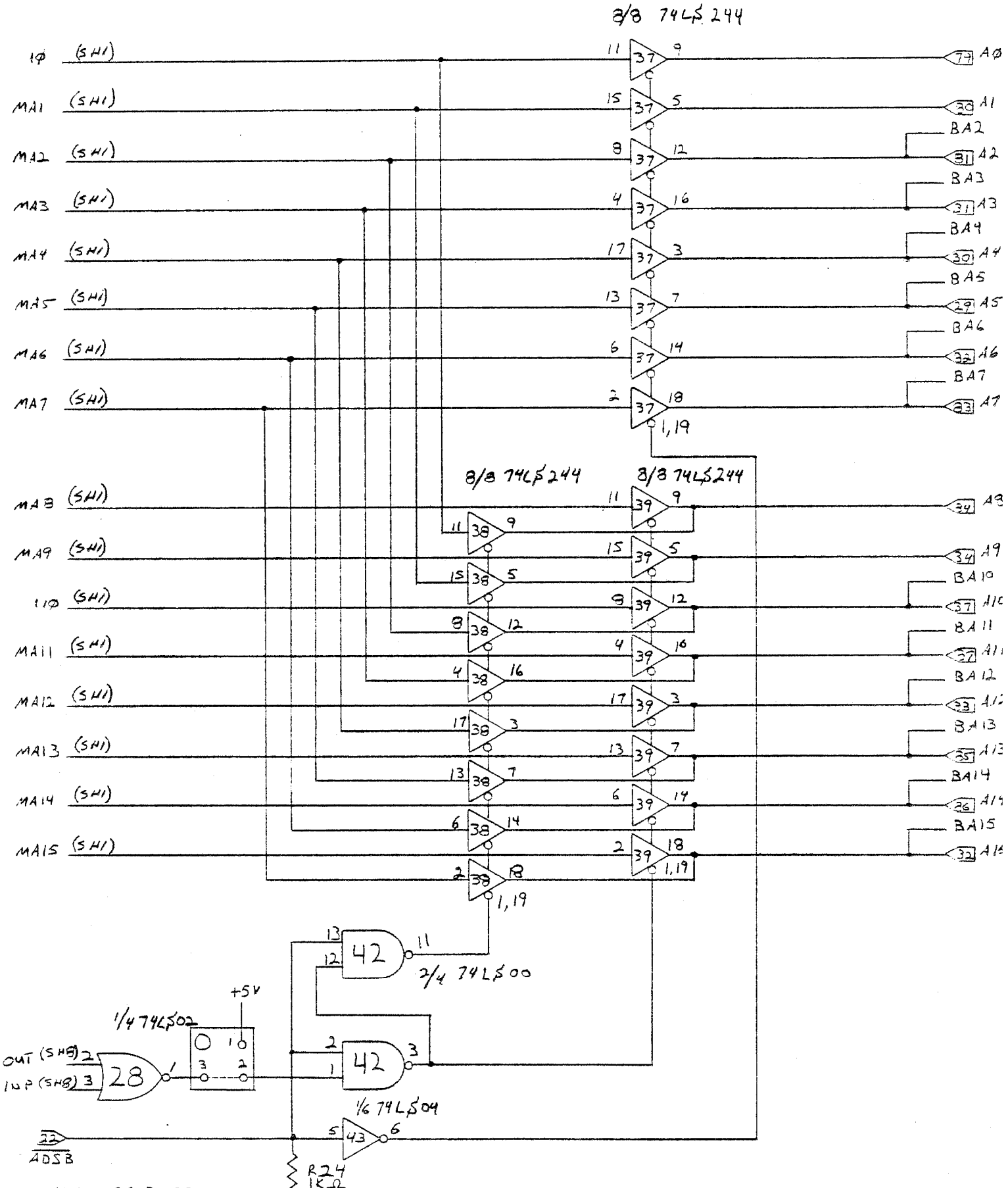
REVISION 2 7/16/80



REVISION 2 7/16/80

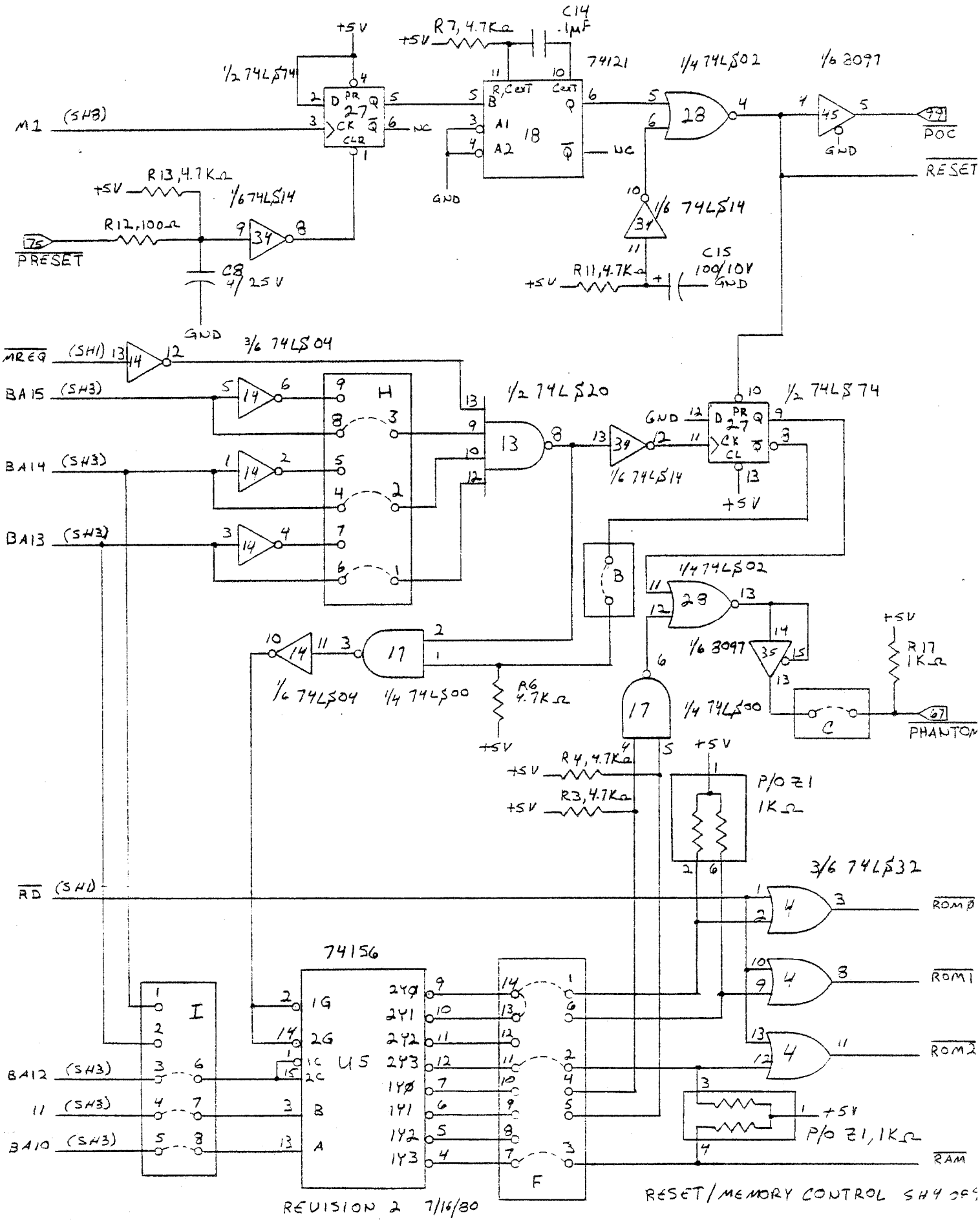
SHEET 2/9

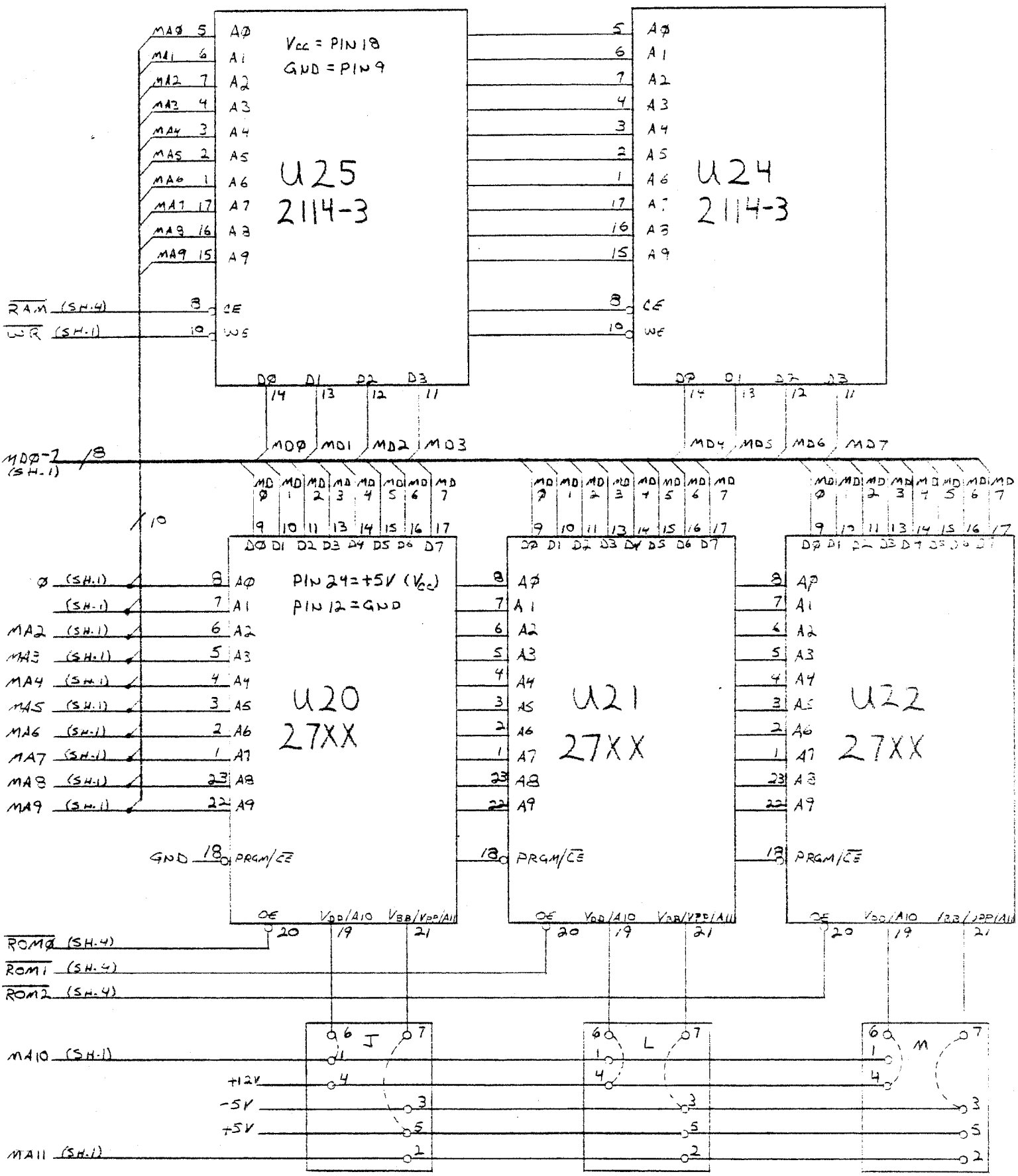
DATA BUS CONTROL



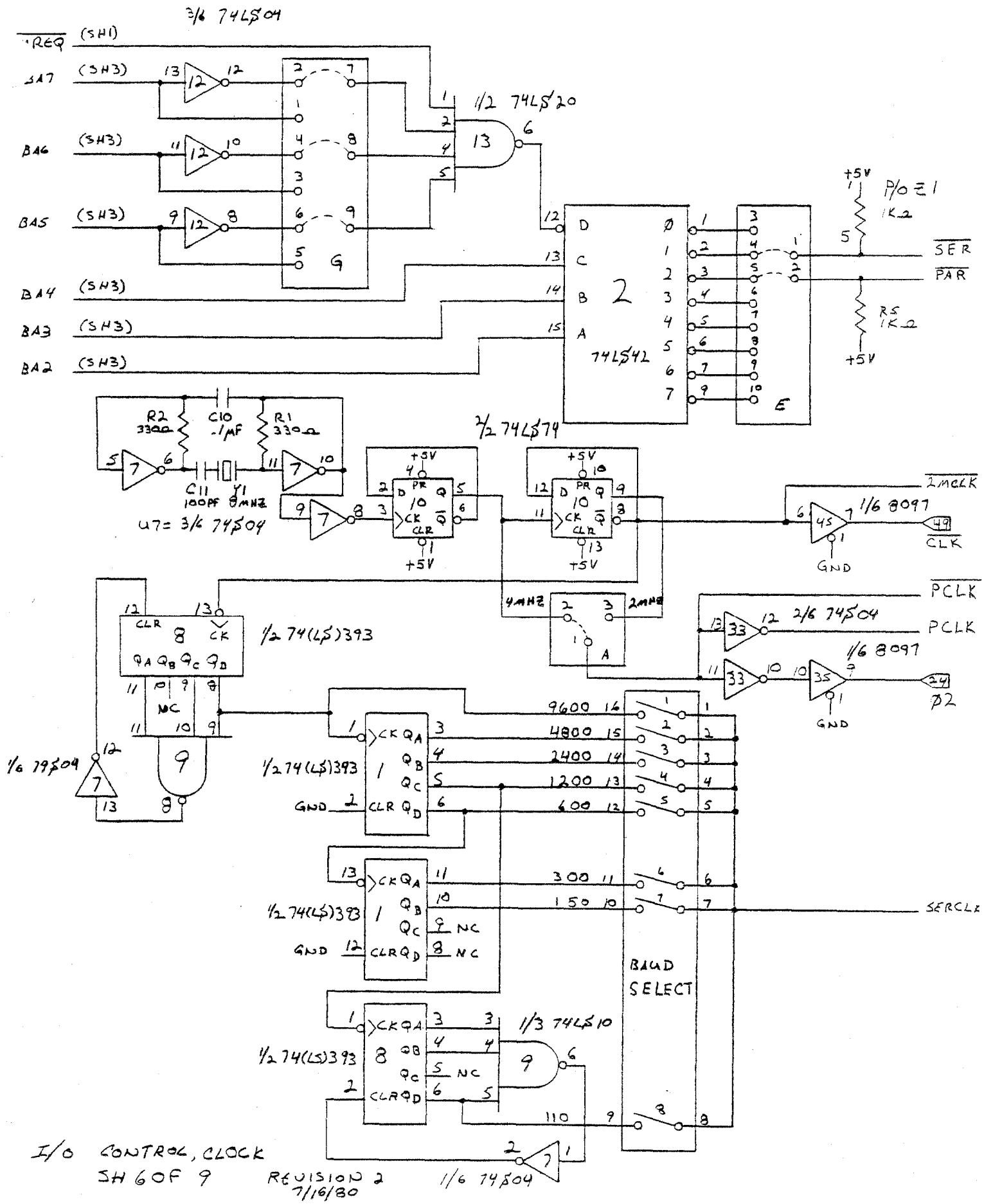
ADDRESS BUFFER,
CONTROL SH3 OF 9

REVISION 2 7/16/80

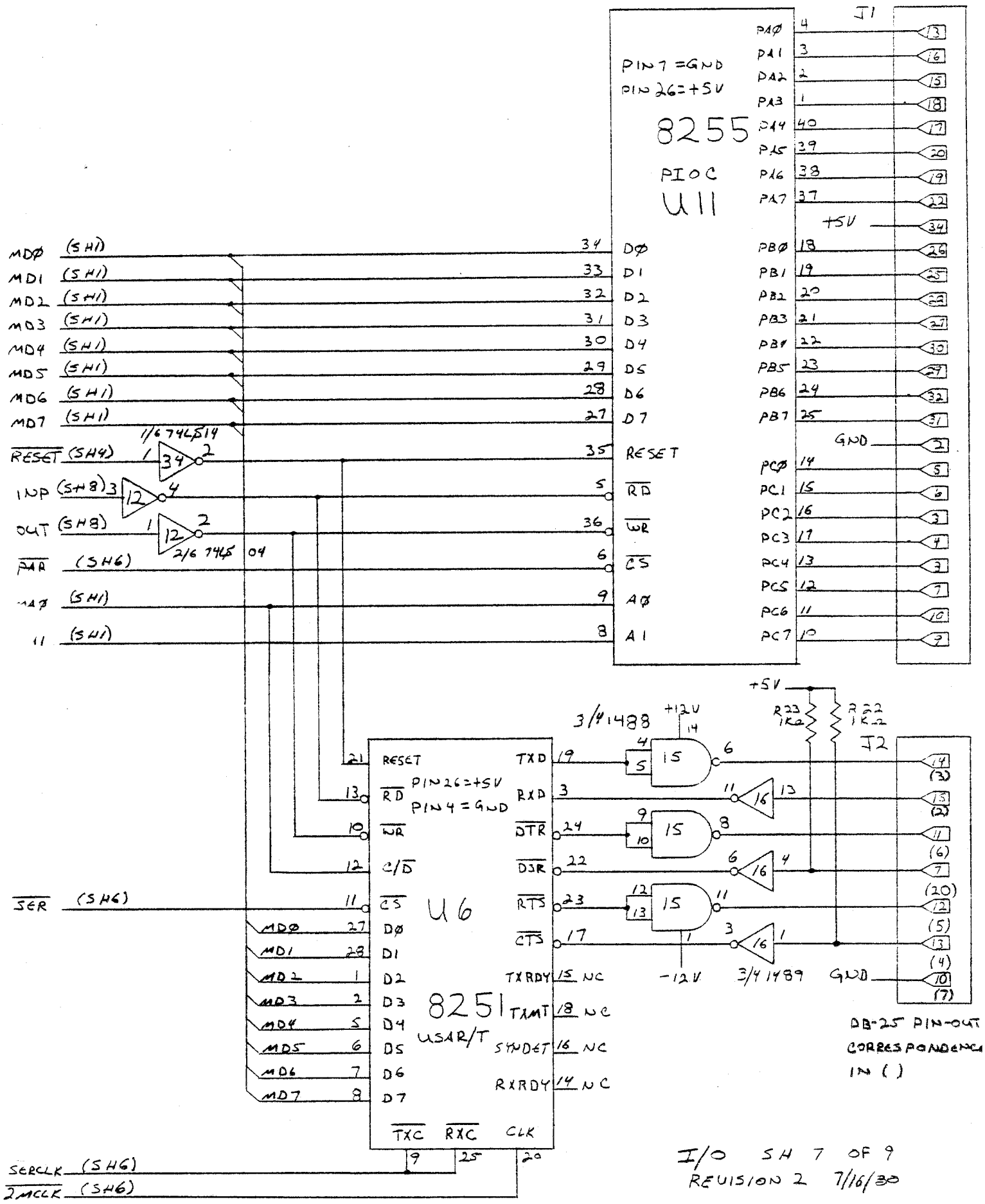

M1 (SH8)
PRESET
R12, 100Ω
R13, 4.7KΩ
+5V
1/6 74LS14
C8 4/25V
GND
1/2 74LS74
74121
R7, 4.7KΩ
C14 .1μF
1/4 74LS02
1/6 8097
POC
RESET
1/6 74LS14
R11, 4.7KΩ
C15 100/10V
MREQ (SH1)
3/6 74LS04
BA15 (SH3)
BA14 (SH3)
BA13 (SH3)
1/2 74LS20
1/2 74LS74
1/4 74LS02
1/6 8097
R17 1KΩ
PHANTOM
1/6 74LS04
1/4 74LS00
R6 4.7KΩ
R4, 4.7KΩ
R3, 4.7KΩ
P/O Z1 1KΩ
3/6 74LS32
RD (SH4)
ROM0
ROM1
ROM2
74156
U5
BA12 (SH3)
11 (SH3)
BA10 (SH3)
P/O Z1, 1KΩ
RAM
REVISION 2 7/16/80
RESET/MEMORY CONTROL SH 4 OF 9



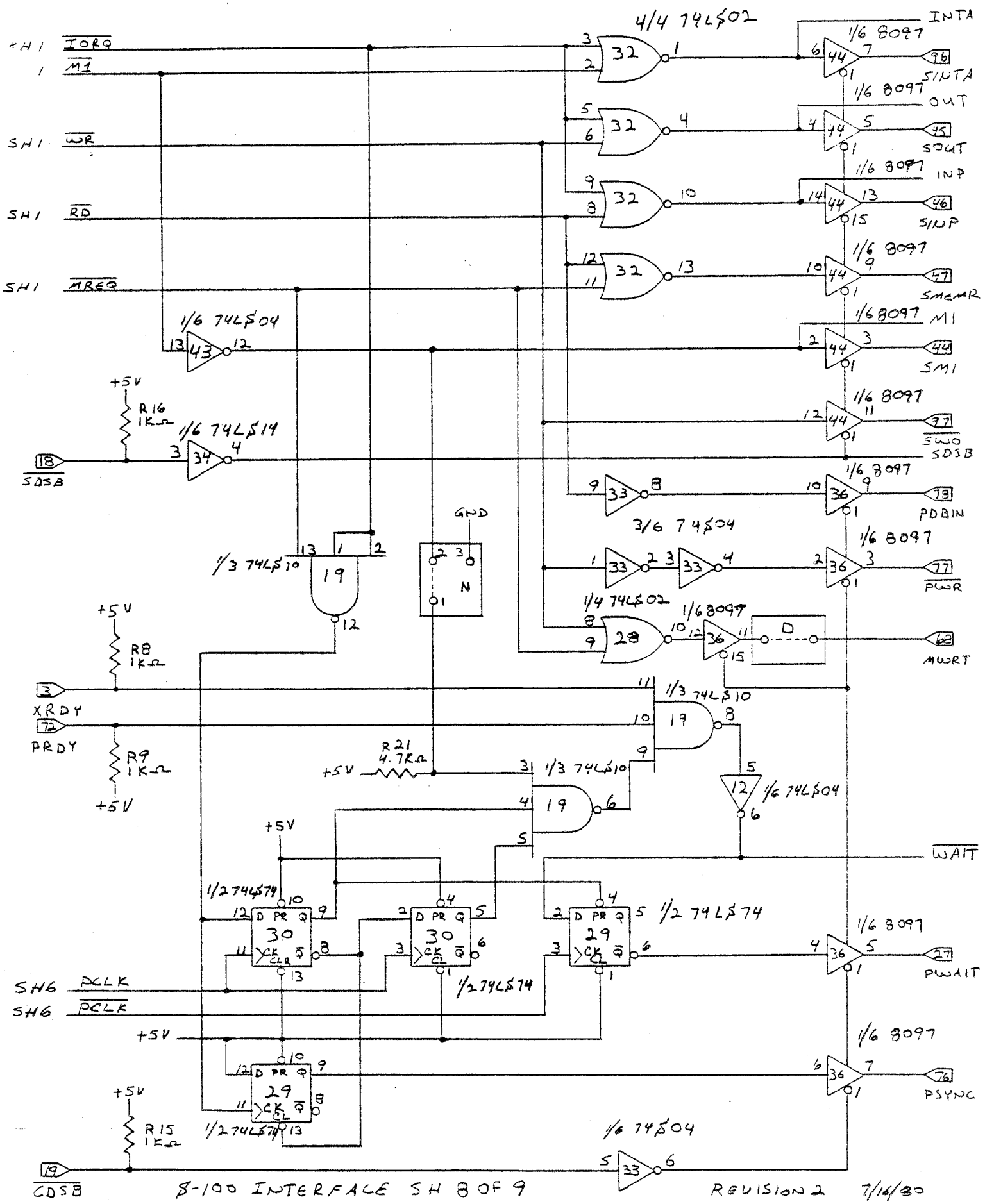
MEMORY SHEET 5/9 REVISION 2 7/16/80



I/O CONTROL, CLOCK
SH 6 OF 9 REVISION 2 7/16/80

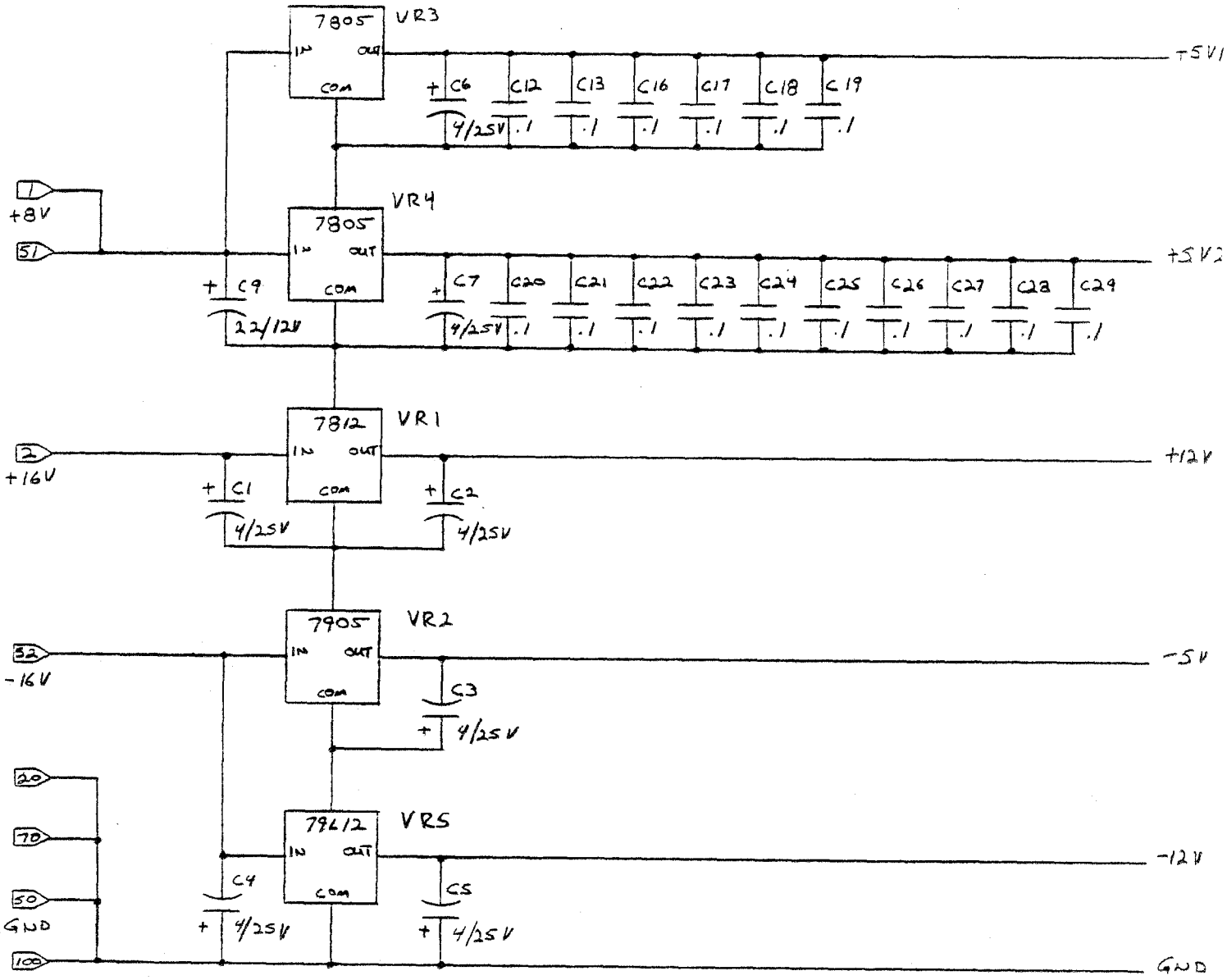


I/O SH 7 OF 9
REVISION 2 7/16/80

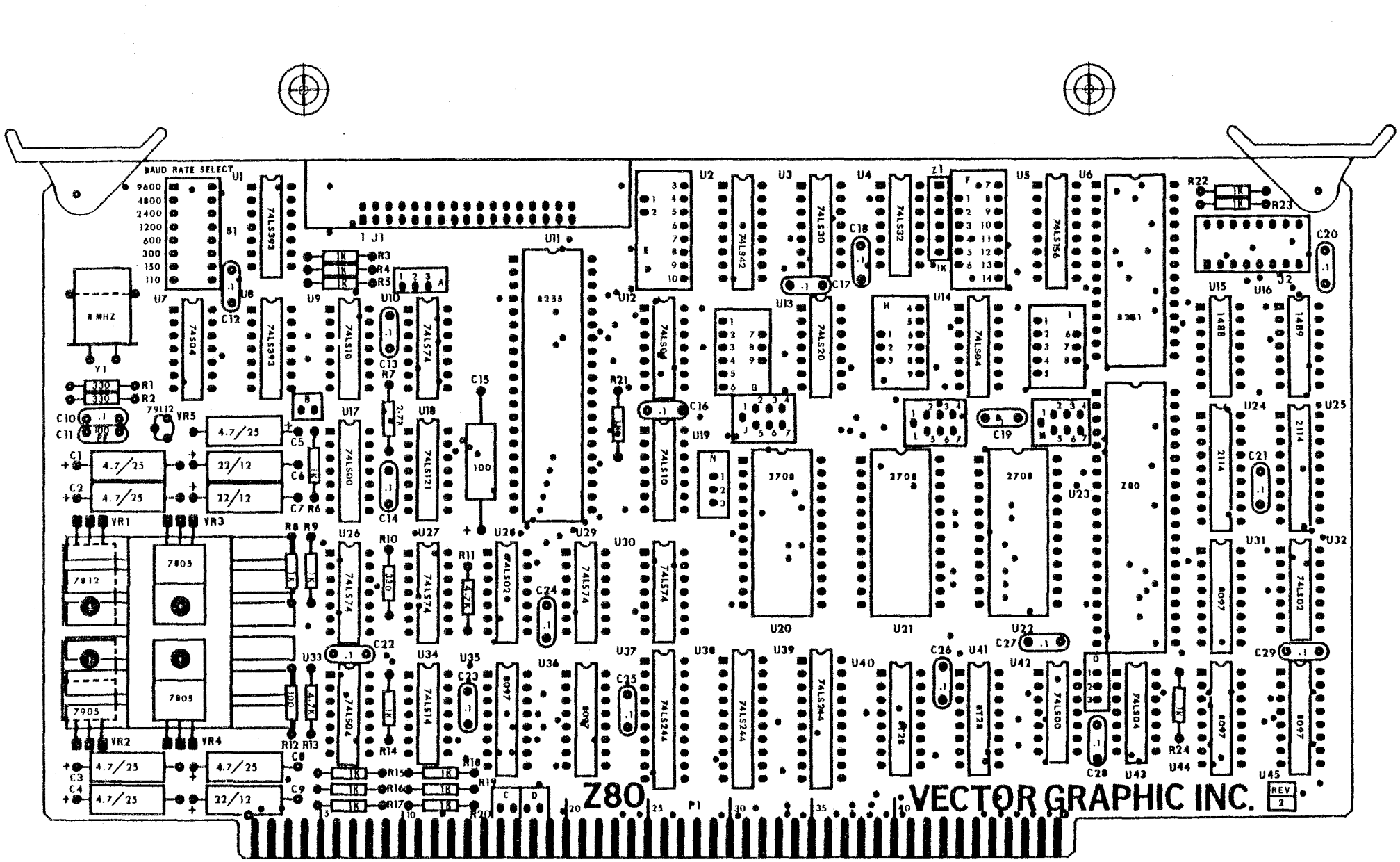

4/4 74LS02
1/6 8097
INTA
SINTA
OUT
SOUT
INP
SINP
SMEMR
M1
SM1
SWO
SDSB
PDBIN
PWR
MWRT
WAIT
PWAIT
PSYNC
SHI IORQ
SHI M1
SHI WR
SHI RD
SHI MREQ
1/6 74LS04
1/6 74LS14
SDSB
R16 1KΩ
+5V
GND
1/3 74LS10
3/6 74S04
1/4 74LS02
XRDY
PRDY
R8 1KΩ
R9 1KΩ
R21 4.7KΩ
1/2 74LS74
SH6 PCLK
SH6 PCLK
R15 1KΩ
CDSB
1/6 74S04
8-100 INTERFACE SH 8 OF 9
REVISION 2 7/16/80

+5V DISTRIBUTION :

+5V1	+5V2
U1-U5	U6
U7-U14	U16
U17-U22	U23-U45



REGULATOR SH 9 OF 9 REVISION 2 7/16/80



NOTES:

1. CAPACITOR VALUES IN MICRO FARADS
2. RESISTOR VALUES IN OHMS

DWN BY: ARTMASTER DESIGN SERVICES	VECTOR GRAPHIC INC. 31364 VIA COLINAS WESTLAKE VILLAGE, CA. 91361
CKD BY:	TITLE: Z80 COMPUTER CARD ASSEMBLY DRAWING
APD BY:	
SHEET: 1 OF 1	DWG. NO. SILKSCREEN — REV. 2