USER'S GUIDE FOR THE

NET/82™ SINGLE BOARD COMPUTER

(PRELIMINARY)

January 4, 1981

SECTION 1.0 - INTRODUCTION

The NET/82 Single Board Computer is another member of the MuSYS family of products designed to give minicomputer throughput rates using microcomputer components. This is achieved by technologies variously referred to as Multi-processing, Distributed Processing, and Networking. While there is some feasibility in a single user having the ability to spin off detached jobs into various processors, the primary demand for this type of architecture arises from requirements for multi-user computers. The two primary reasons leading to a decision in favor of a multi-user computer are a desire to minimize the need for costly peripheral equipment, such as hard disks and letter quality printers, and a need to access shared data bases, usually in real time.

It can truly be said that with today's semiconductor technology, the Central Processing Unit (CPU) and it's memory are the cheapest part of a modern data processing system. In fact, hardware in general is becoming so inexpensive that the entire thrust of system implementation is changing. Because software costs are the largest part of any business computer installation of any size, the requirements of the software become the prime factors in the design and selection of the hardware. Thus, one of the basic design criteria for new systems is to try to take advantage of an existing body of software, while making available the benefits of a new hardware technology.

To this end, the NET/82 has been specifically designed to support the largest body of microcomputer software available today, the software which runs on 8080/8085/Z80 based computers. Features have been incorporated which will allow a number of different operating systems and applications software packages to be supported. These features include bank switched memory, floating point processor support, and full interrupt control. All of these features are required for the most advanced systems, but are either inexpensive or optional if they are not required for a specific application. If you investigate the matchup between your application and the features on the NET/82, or the other boards in the same family, you will find exactly what you need, proving that the MuSYS product line is the optimum hardware selection for users desiring compatibility with existing software, but needing greater throughput and flexibility than is achievable with traditional single-processor hardware.

The NET/82 is a complete computer on a single board, with the exception that there is no provision for supporting local disk devices. In the multi-processor architecture for which the NET/82 is designed, disk operations are controlled by another computer, which is accessed through a communications protocol of some sort. Since the NET/82 is designed as an I/O mapped slave according to the IEEE-696 specification for S-100 computers, the usual communications method is to exchange data with the bus master processor controlling the local S-100 bus. That processor may, in turn, be part of a larger communications network, making an almost unlimited amount of data and external devices available to the user and software associated with an individual NET/82.

Other uses are possible, using either the NET/82 or other boards from the same family. Once a distributed processing architecture has been established, large increases in overall system throughput may be achieved by allocating individual functions to individual processors. Thus, a processor may be dedicated to running a single high-speed printer as an intelligent print de-spooler, or it may run several slower printers in the same manner. Other processors in the same box may be acting as intelligent communications front-ends, providing high-speed access to the peripherals in other boxes of various types. Still other processors may be dedicated to performing various tasks which distribute the workload of the overall system, such as background batch processing or data base management. The real power of this architecture will only be felt when the applications developers include provisions for distributed processing within their applications. Some progress in this direction is being made, with at least one major vendor of data base management software announcing a multi-processor data base management system which resides in one computer, while servicing inquiry/update requests from data base users residing in other computers.

The NET/82 is designed to be a flexible, low cost, unit computer, capable of being networked with other computers to meet applications needs similar to those discussed above. Almost anything achievable by computer is possible, if you only have enough of these computers working in parallel. So the last part of this discussion is a challenge to the user. Let us hear about your novel and unique applications for these hardware building blocks. If we feel there would be enough general interest, we will see to it that your idea is published in an appropriate place. Let your imagination run free, and the world will benefit.

(This page was intentionally left blank)

SECTION 2.0 - BOARD SET-UP AND INSTALLATION

The NET/82 board will normally be delivered pre-configured for the operating system environment you specified in your order. Under these circumstances, the only set-up required is the setting of the board address switches to the proper I/O address for the next network processor board in your system. More advanced users may require additional set-up, such as field installation of optional features and re-jumpering of the various jumper areas on the board. The purpose of this section is to explain these various hardware options.

SECTION 2.1 - ADDRESS SWITCHES

The address switches are comprised of sections 1 to 7 of the switch SW1, which is located in the lower left hand part of the board. The easiest way to set the switches is to turn the board upside down so that the switches may be read in order, 1 to 7. Switch 1 is the most significant bit (MSB) of the board address, corresponding to address line 7 (A7) on the S-100 bus. Switch 7 is the least significant bit (LSB) of the board address, corresponding to address line 1 (Al) on the S-100 bus. Only seven switches are used because the NET/82 requires two I/O ports out of the system address space, using address line 0 (AO) to select between them.

Each position of the switch is placed into a one or zero state by pushing in on the appropriate end of the switch. Use a small pointed object for this operation. When the switch is closed (side next to the S-100 connector pushed in) a zero is established for that bit position. When the switch is open (side away from the S-100 connector pushed in) a one is selected for that bit position.

You must determine the proper I/O port address required by your operating system software. Once this is known, break it down to the binary bits represented by bits A7 to Al of the address. Then set each switch position, as described above.

Below is a diagram of a switch set for address 70 hex, with the sense switch (described below) in the open (one) position:

```
_________________
|1 2 3 4 5 6 7 8|
|               | NOTE: An asterisk (*) denotes the switch
|*       * * *  |       pushed in at that point.
|  * * *       *|
|-----OPEN------|
```

SECTION 2.2 - SENSE SWITCH

Section 8 of the switch is the sense switch. The state of this switch may be read by the software, and used to control program execution, when used with the standard EPROM supplied by MuSYS, the sense switch selects normal bootstrap mode, or board diagnostic mode. Switch section 8 should be open for normal operation, and closed to invoke diagnostics. Note that the sense switch is wired in parallel with J2 pin 4, so that diagnostics may also be invoked from an external source, such as a front panel. Closing section 8 of the switch is equivalent to grounding J2 pin 4, and either one will cause a low (zero) to be sensed by the software.

SECTION 2.3 - INTERRUPT JUMPER AREA

The interrupt jumper area is in the lower left-hand part of the board, immediately to the left of the address switches. It consists of eight sections of five pins each, with a section corresponding to one of the eight vectored interrupt lines from the S-100 bus. The sections are numbered from 0 to 7, corresponding to VI0* to VI7*, with section 0 as the highest priority and section 7 as the lowest, in fixed priority mode. (See the section on interrupt controller programming).

Each section has five pins, indicated as rows A through E on the board. Rows A and C are local (on-board) interrupt sources. Row B is the input to the interrupt controller chip. Row D consists of signals which may be used to generate interrupts to the S-100 bus. Row E is the S-100 interrupt lines, VI0* to VI7*. The S-100 bus signals may be wired either as inputs (to the interrupt controller, row B) or as outputs (normally from row D, but the signals from rows A and C could also be used). The signals are arranged so that the most likely connections may be made using push-on jumpers, but wire-wrap wire may also be used, as necessary. Below is a diagram of the interrupt jumper area:

```
A    *       *       *       *       *       *       *       *
   WRSTAT  DIENA   DOENA                           END12

B    *       *       *       *       *       *       *       *
   IREQ0   IREQ1   IREQ2   IREQ3   IREQ4   IREQ5   IREQ6   IREQ7

C    *       *       *       *       *       *       *       *
   WRD01   WRD02   WRD03   64PPS   PERROR  SVREQ   END11   1PPS

D    *       *       *       *       *       *       *       *
     ---SB5---       ---SB6---       ---SB7---       -REQUEST-

E    *       *       *       *       *       *       *      *
    VI0     VI1     VI2     VI3     VI4     VI5     VI6    VI7

     0       1       2       3       4       5       6       7
```

a. Interrupts Generated by the S—100 Bus Master. The computer controlling the S-100 bus may generate a number of interrupts to the NET/82 board. WRSTAT, DIENA, and DOENA are generally used with the no-wait protocol (see the section on inter-processor communication). WRSTAT is asserted when the bus master writes a byte to the NET/82 Command/Status port. DIENA is asserted when the master reads a byte from the NET/82 data port. DOENA is asserted when the master writes a byte to the NET/82 data port.

The remaining three signals are special cases of the WRSTAT signal. In each case, WRSTAT is logically ended with the appropriate data bit sent by the master in the byte output to the NET/82 Command/Status port. Bits 1, 2, and 3 in the output byte generate the signals WRD01, WRD02, and WRD03 respectively.

b. Floating Point Processor Interrupts. If a floating point processor is installed on the NET/82 board, it may be used to generate interrupts when servicing is required or an operation is complete. The signal SVREQ is a request for the next data byte in a multiple byte transfer to or from the floating point processor. The signals END11 and END12 are generated when the operation is complete and the result is ready. END11 and END12 are opposite polarities of the same signal, and are provided for compatibility with the two types of floating point processors which are usable on the NET/82. The END11 signal is used for an AMD 9511 or an Intel 8231. The END12 signal is used for an AMD 9512 or an Intel 8232.

c. Other Local Interrupt Sources. The PERROR signal is generated when the NET/82 memory parity checking circuitry detects a parity error on a byte read from memory. This signal may be used to initiate any appropriate software error recovery procedures.

The signals 64PPS and 1PPS are generated from the third channel of the 8253 timer chip, and will only be accurate as shown when that channel is initialized to generate the 256Hz square wave required for the hardware divider. This divider takes the output of the 8253 and divides it by 4 to get the 64PPS signal, and by 256 to get the 1PPS signal. If the 8253 is set up for a rate other than 256Hz, the 1PPS and 64PPS signals will change rates appropriately.

d. Interrupts to the S-100 Bus. The signals REQUEST, SB5, SB6, and SB7 are generated locally by the NET/82 if a one is written into the appropriate bit of the local Command/Status port. These signals will normally be used when the software in the NET/82 desires to generate one of the signals VIO through VI7 to interrupt the master. However, since the signals VIO-7 may also be interrupts into the NET/82, these signals may also be used to interrupt another NET/82.

SECTION 2.4 - CLOCK JUMPER AREA

There are four clock options on the NET/82. Each consists of three pins which are part of the JA - RATE SELECT jumper area located between the SIO and the 8253 (U42 and U41). The leftmost three pins select the clock rate for the floating point processor chip, U13. Connecting 1-2 selects a 4MHz clock rate while connecting 2-3 selects a 2MHz clock rate. The rate selected must be equal to or less than the maximum clock rate specified for the floating point processor chip you have chosen.

The remaining 9 pins of the JA - RATE SELECT jumper area are used to select the input clock rate for the three channels of the 8253 programmable timer. Pins 4-6 are for channel 0 (the left serial port, port A), pins 7-9 are for channel 1 (the right serial port, port B), and pins 10-12 are for channel 2 (the Real Time Clock, which generates the 64PPS and 1PPS interrupt signals). Since the maximum clock rate acceptable to the 8253 is 2.5MHz, the two clock sources are divided down to meet this specification. Therefore, the left pin of each section is 2.4576MHz (4.9152MHz divided by 2) while the right pin is 2MHz (8MHz divided by 4).

The standard jumper option for JA - RATE SELECT is to have each section jumpered left-to-center, selecting the 4MHz clock for the floating point processor and the 2.4576MHz clock for the 8253 inputs. Starting with revision B of the NET/82 board, these selections will be in the etch, and no jumper pins will be provided. If it is necessary to change the configuration of a revision B or later board, simply cut the etch (on the rear of the board) which is connecting the left and center pads and wire a jumper between the right and center pads.

SECTION 2.5 - EPROM SELECTION

The NET/82 is supplied with a standard 2716 EPROM programmed with the boot sequence for your operating system. This EPROM also contains the diagnostic software which may be invoked by asserting the sense line (J2 pin 4 or section 8 of SW1) to a zero (low, or ground) state. If it is necessary to replace this EPROM, please note that a device with an access time of 350ns or faster is required in order to meet the timing requirements during the M1 cycle of a 4MHz Z80A processor. There is no provision in the NET/82 for generation of a wait state when the EPROM is selected.

Please note that the Texas Instruments 2716 EPROM is incompatible with all of the other manufacturers of 2716s, and may not be used with the NET/82. However, the Texas Instruments

2516 EPROM may be used in place of a 2716. Simply select 2716 as the EPROM type for the NET/82.

The other types of EPROMs which may be used are the 2532 and the 2732. The jumper areas for selecting the EPROM type are located immediately to the right of the EPROM itself (U44). Two jumpers are required, one on each area. Each jumper connects horizontally the two pins appropriate to the type of EPROM being used. The three selections are clearly labeled for 2716 (or 2516), 2532, or 2732. Note that in all cases, the EPROM is decoded as a 4K byte address space. Changing the EPROM type selection does not affect the amount of memory overlaid by the EPROM. When a 2K byte EPROM (2516/2716) is used, it will appear twice at the bottom end of memory (addresses 0000H to 0FFFH).

For revision B or later boards, the EPROM type has been selected in the etch as a 2716 (or 2516) type EPROM. If it is necessary to change the type of EPROM on these boards, cut the etch (on the rear of the board) and jumper the appropriate pads for the EPROM you intend to use.

SECTION 2.6 - ASYNCHRONOUS NULL-MODEM PADDLE BOARD JUMPERS

As a standard feature, the NET/82 board is supplied with two asynchronous null-modem paddleboards. These boards are designed to interface the serial ports of the NET/82 to most standard terminals or printers using standard 'straight through' cables. This means that ribbon cables which wire all lines in parallel are suitable for use with these boards. No interchange of pins 2 and 3, or any other special modification of the cable should be required. Other types of paddle boards are available as options, should you have special requirements.

The asynchronous null-modem paddleboard has three jumper options. For purposes of this discussion, take the paddle board and orient it so that the DB-25 (female) connector is pointing up and the 16 pin header (male) is pointing toward you.

Located immediately above pin 8 of the MC1489A (75188A) are two pads. In most cases, these are left unconnected, as shipped from the factory. However, if it is necessary to connect pin 1 of the RS-232 cable (frame ground) to the computer ground, it may be done by connecting these two pads. Note that this should only be done on advice from a competent individual, as it may impact compliance with local electrical codes.

To the right of the above area, and approximately above pins 1-6 of the 16 pin header, is the remote reset option jumper area. In most cases, this will be wired center to left. Starting with revision B of the paddle board, this area will be wired this way in the etch, and it will be necessary to cut the etch on the rear of the board in order to change the configuration of this area. when wired center to right, the remote reset option is enabled. Note that this is only effective for the paddleboard connected to the left serial port (J3, port A) on the NET/82 board. The remote reset option logically ties the computer's reset input to the RS-232 pin selected by the third jumper area (see below). Usually, that jumper area will be set up to select pin 19, and a reset will be obtained by shorting RS-232 pins 18 and 19. Pin 18 is supplied as a current source (through a 1K resistor) for this purpose. Whichever RS-232 pin is selected, the computer will be reset when that line goes low (inactive, negative voltage). Terminals which normally assert Data Terminal Ready on pin 20, but which can cause pin 20 to go inactive under operator control, may be used to initiate a reset in this manner. The IBM 3101 is an example of such a terminal. In this case, pin 20 should be selected below. Whichever pin is selected, a resistor on the paddle board will pull that line up, ensuring that a powered-off terminal will not hold the computer in a reset condition. Note that the remote reset feature should only be used if the software operating system has the ability to recover from a hardware reset issued at any time. Otherwise, the processor may be unusable after it has been reset.

Further to the right, above and to the left of pin 14 of the MC1488 (75188), is a six pin jumper area which is used to select the handshake/reset pin. The middle two pins are common, and should be wired to one of the four outside pins, usually using a push-on jumper plug. The area may be used to select between pins 11, 14, 19, and 20 of the RS-232 cable. {Note that pin 11 is only available on revision B and later versions of the paddle board). Pin 11 is usually used for Texas Instruments 800 series printers, and other compatible devices. Pin 14 is usually used by Lear Siegler 300 series printers. Pin 19 is usually used by NEC Spinwriter compatible printers. Pin 20 is usually used at all other times. The area may be left completely open (no jumper at all) with no effect, as a pull-up resistor is present to ensure operation with an open cable. In such a situation, the remote device appears as ready all of the time. Below is a diagram of this jumper area:

```
Pin 19 — > * * * <— Pin 20
Pin 14 — > * * * <— Pin 11 (Rev. B or later)
```

(This page was intentionally left blank)

SECTION 3.0 - PROGRAMMING

Normally, you will have received the software necessary to use your NET/82 either from MuSYS or another software vendor. If you are writing your own software for one or more of the features on the NET/82, this section will guide you in preparing that software. There are six devices which require programming:

a. The Z80A-SIO/2, which supports 2 serial channels.

b. The floating point processor.

c. The 8253 Programmable Interval Timer (for baud rate and real-time-clock).

d. The 9519 Interrupt Controller.

e. The memory bank select logic.

f. The Inter-Processor Communication Channel.

Data sheets for the first four items, including both types of floating point processors, appear in appendices in the back of this manual. No attempt will be made here to explain the detailed programming of these devices, beyond commenting on the board dependent parameters of each of these chips. The memory bank select logic and the Inter-Processor Communication channel will be discussed in detail below.

SECTION 3.1 - I/O PORT ADDRESSES

The I/O ports used on the NET/82 are listed below, with all addresses in hexadecimal:

PORT	DESCRIPTION
00	SIO channel A, Data (J3, the left connector)
01	SIO channel A, Command/Status
02	SIO channel B, Data (J4, the right connector)
03	SIO channel B, Command/Status
08	Floating Point Processor, Data
09	Floating Point Processor, Command/Status
10	8253 Channel 0, Data (SIO channel A, baud rate)
11	8253 Channel 1, Data (SIO channel B, baud rate)
12	8253 Channel 2, Data (Real-Time-Clock interrupt)
13	8253 Command
18	Interrupt Controller, Data
19	Interrupt Controller, Command/Status
1D	Memory Bank Select, Command
1E	Inter-Processor Communication, Data
1F	Inter-Processor Communication, Command/Status

SECTION 3.2 - INTERRUPT USAGE

The NET/82 interrupt daisy chain contains only the SIO and the 9519 interrupt controller. Due to the characteristics of the SIO, only interrupt mode 2 is permissible.

SECTION 3.3 - SIO PROGRAMMING

The NET/82 uses the Z80A-SIO/2, which has symmetrical pin-outs for channels A and B. (SYNCA is wired as a no-connect). This means that both channels may be used in high baud rate applications, which require separate clocks for transmit and receive. The actual characteristics of the serial port are determined by an external paddle board, which contains the logic necessary to adapt the SIO pin-outs to a particular application. The NET/82 is configured so that the Wait/Ready lines on the SIO may be used in the Wait mode. This means that block I/O instructions may be used for high baud rate applications. The SIO is the highest priority device on the interrupt daisy chain. The Z80 must be run in mode 2 when using the SIO in interrupt mode. For further details on SIO programming, refer to the appendix which contains the SIO data sheet.

SECTION 3.4 - FLOATING POINT PROCESSOR PROGRAMMING

Normally, the floating point processor will be run in programmed I/O mode, under control of the applications program. Block I/O instructions to the floating point processor are acceptable because the PAUSE* output is wired into the NET/82 wait state generator. If you are using interrupts with this device, be sure that the interrupt selection {END11 or END12) matches the type of chip you are using. The use of 4MHz parts is highly recommended, as the increase in throughput more than makes up for the slight increase in cost. After all, increased speed is the reason for buying a floating point processor in the first place. The appendicies contain data sheets on the two Intel 4MHz devices (8231 and 8232). Refer to these for programming information.

SECTION 3.5 - INTERVAL TIMER PROGRAMMING

The 8253 Programmable Interval Timer is used primarily in mode 4, as a square wave rate generator. Channel 0 generates the clock signal for SIO channel A, channel 1 generates the clock for SIO channel B, and channel 2 should generate a 256Hz signal to drive the real-time-clock. The input clock rate for each channel is selectable as either 2.000MHz or as 2.4576MHz (See section 2.4). Normally 2.4576MHz is used. The table below gives the divisors for commonly used baud rates:

BAUD	CLOCK		
RATE	RATE (16x)	DIVISOR	
--	256HZ	9600	(For channel 3, RTC)
50	800Hz	3072	
75	1200Hz	2048	
110	1760.5Hz	1396	(slightly over speed)
134.5	2152HZ	1142	
150	2400Hz	1024	
300	4800Hz	512	
600	9600HZ	256	
1200	19.2KHz	128	
1800	28.9KHz	85	(slightly over speed)
2000	31.9KHz	77	(slightly under speed)
2400	38.4KHz	64	
3600	57.2KHz	43	(slightly under speed)
4800	76.8KHz	32	
7200	117KHz	21	(somewhat over speed)
9600	153.6KHz	16	
19.2K	307.2KHz	8	

SECTION 3.6 - INTERRUPT CONTROLLER PROGRAMMING

The 9519 Interrupt Controller is the second, and final, device on the interrupt daisy chain. Therefore, SIO interrupts will take priority over 9519 interrupts. The hardware is designed so that all interrupt inputs are asserted low, and are pulled high when not active. It should never hurt to enable interrupts which are not wired to anything. Since many interrupt signals cannot be controlled by the NET/82 software, edge triggered mode is generally called for. To simplify programming of interrupt service routines, auto-clear is usually enabled for all interrupting channels. The 9519 may be used with any Z80 interrupt mode, however, due to limitations in the SIO, mode 2 will normally be used. In any case, responses are limited to one byte, as no logic exists on board to correct for multiple byte interrupt acknowledge cycles. Selection of the interrupting devices is covered in section 2.3, and actual programming of the 9519 is contained in a data sheet in an appendix.

SECTION 3.7 - MEMORY BANK SELECT PROGRAMMING

The memory bank select circuitry offers one of the most versatile bank switching schemes available today. The design is based on the assumption that some smaller amount of memory (1K to 16K) must remain common (on all of the time), while a larger amount of memory (48K - 63K) is switched. The switchable memory is at the bottom of the address space, while the common memory is measured from the top down. The dividing line between the common and switchable memory is determined by software. The command port for the bank select logic (port 1DH) takes an output byte where the upper four bits (7-4)define the bank switching boundary and the lower two bits (1-0)define which switchable bank, if any, is to be switched on.

The table below defines the value sent to the upper four bits of the command port, the amount of switchable memory, and the first byte of un-switched memory, above the boundary:

VALUE (bits 7-4} (hex)	BANK SIZE	UNSWITCHABLE BOUNDARY (hex)
0	48K	C000
1	49K	C400
2	50K	C800
3	51K	CC00
4	52K	D000
5	53K	D400
6	54K	D800
7	55K	DC00
8	56K	E000
9	57K	E400
A	58K	E800
B	59K	EC00
C	60K	F000
D	61K	F400
E	62K	F800
F	63K	FC00

The table below defines the value sent to the lower two bits of the command port, and the resulting bank which is selected:

VALUE	BANK
0	None
1	0
2	1
3	Illegal (0 selected)

SECTION 3.8 - INTER-PROCESSOR CHANNEL PROGRAMMING

The inter-processor communication channel is the device which makes the NET/82 useful in a multi-processor environment. The NET/82 is designed with a special high-speed, discrete logic, programmed I/O communication channel. This logic makes it possible to transfer data at near the maximum speed possible with Z80 block I/O instructions, ensuring coordination between processors by generating wait states in the slave and deliberately programming the master to execute slower than the slave. This gives the fastest data transfer rates possible, short of an expensive and unreliable DMA channel for each slave.

Programming of the inter-processor channel is similar to the programming used on the MuSYS NET/80™ board, for those familiar with that product. Most of the concepts are the same, although the detailed design has been considerably improved. In addition to the normal mode of synchronized transfers, the NET/82 will also support a no-wait protocol. The no-wait protocol allows one byte at a time to be exchanged between processors, on an interrupt driven basis. Also, when the no-wait protocol is in effect, command bytes written to the command port by the master are readable by the slave. This may be particularly useful if the master sends a general interrupt, but wishes some specific action to be taken. Your software may use either mode, or a combination of them.

While the following discussions are broken down into a master and slave side view, you must realize that both sides have to work together for the whole thing to work. It would be highly beneficial for the same programmer to work on both sides of the interface.

SECTION 3.8.1 - SLAVE SIDE PROGRAMMING

The software running in the NET/82 sees two I/O ports when communicating with the master. The data port (1EH) is eight bits of read/write data, which are separately latched (in LS373s). An output to the data port loads a byte into the output latch and an input from the data port reads the contents of the input latch. If no-wait protocol is not selected, accessing the data port for either a read or write will cause the NET/82 to go into a wait state until the master accesses the other side of the data port. This wait-state protocol is the normal mode of operation.

The command port (1FH) is a bit sensitive port which receives a command byte from the NET/82 software. The bits are:

BIT	FUNCTION
7,6,5	Arbitrary command bits. Readable by the master, usable as interrupts to the master (or other slaves).
4	1 = No-Wait protocol, 0 = Wait-State Protocol (normal)
3	Request bit. Same as arbitrary command bits, but usually used to indicate slave requests. Also can be read by the NET/82, which can look for it to clear.

NOTE: Bits 7 to 3 are clearable by the master.

2	1 = Clear Overrun Status.
1	1 = Clear Parity Error Status.
0	1 = Enable EPROM at addresses 0000-0FFFH. (0 = disable)

The status port (1FH) is a bit sensitive port which NET/82 software may read to determine the status of certain signals. The bits are:

BIT	FUNCTION
7	0 = Ring, channel B (from certain modem paddle boards only).
6	0 = Data Set Ready, channel B
5	0 = Ring, channel A (see above)
4	0 = Data Set Ready, channel A
3	1 = Request set (echoes command port bit 3)
2	1 = Overrun detected (see discussion on overrun)
1	1 = Parity error detected
0	0 = Sense switch closed (or J2 pin 4 grounded)

The Data Set Ready and Ring signals will reflect the signals presented to the NET/82 by the paddle board plugged into the respective connectors, J3 for channel A, and J4 for channel B. The request bit is readable in the status port so that the NET/82, software may detect the fact that the master has cleared bits 3 to 7 of the command port. Overrun is discussed later. Parity errors are detected by the memory parity circuitry if any byte is read with even parity. Odd parity is always generated on memory writes. The standard use of the sense switch is to invoke the EPROM resident diagnostic package after a board reset. The EPROM detects a zero in bit zero to start diagnostics.

I/O transfers between the master and slave may be broken down into three stages. The set-up stage establishes the protocol and the direction of transfer. This stage normally uses the bits in the command/status register to communicate with the master. The transfer stage exchanges one or more bytes of data between the processors in either the no-wait or wait-state mode. The results stage checks the status and validates that an error free transfer occurred. Normally, this means a check of the overrun status, since the wait-state protocol is usually used.

The NET/82 software must be written according to which interrupts are to be used and what the overall protocol is. A discussion of an example protocol appears later, after an explanation of how the NET/82 appears to the master.

SECTION 3.8.2 - MASTER SIDE PROGRAMMING

The software running in the S-100 bus master also sees two I/O ports when communicating with the NET/82. The data port (even) is eight bits of read/write data. Reads and writes to the data port communicate with the other side of the LS373 latches that the slave sees on it's data port. No wait state is ever generated for the master, so the software in the master must be written to execute slowly enough to properly communicate with the NET/82. This may require some conditional assembly code to customize according to processor speed in the master. This all assumes the normal wait-state protocol. Where no-wait protocol is used, processor synchronization is via flags or interrupts, so there should be no sensitivity to processor speed variations.

The command port (write to odd) is a bit sensitive port which receives a command byte from the master software. Only the 4 least significant bits have meaning. These are defined as:

BIT	FUNCTION
3	Clear bits 7-3 of the NET/82 local command register. (These bits appear in the status register, below). Also, generate the interrupt WRD03, if enabled.
2	Generate the interrupt WRD02, if enabled.
1	Generate the interrupt WRD01, if enabled.
0	1 = Reset the NET/82, 0 = Enable the NET/82 to run.

The reset command bit is latched by an on-board flip/flop, so the master must first output a one in the reset bit, wait for a short delay, then output a zero to enable the NET/82 to run. Note that a reset to the NET/82 may destroy some or all of the dynamic memory contents. This would be especially true, if the reset is left asserted for any length of time, as no refresh signals are generated during reset. However, it is not unusual for one byte to be destroyed by the reset itself due to aborting a memory cycle in mid stream.

The interrupt signals go to the interrupt jumper area on the NET/82, where they may be fed into the 9519 interrupt controller. The software in the NET/82 is responsible for setting up the NET/82 so that interrupts are properly latched, recognized, and processed. The signal WRD03, in addition to generating an interrupt, will clear bits 7-3 of the NET/82 command register. The NET/82 may monitor this action, and this may be used as part of a communication protocol between the processors.

The status port (read from odd) is a bit sensitive port which the master software may read to determine the status of an individual NET/82 board. The bits are:

BIT FUNCTION

7,6,5 Arbitrary command bits, from the NET/82 local command register (slave I/O port 1FH).

4 1 = No-Wait protocol, 0 = Wait-State protocol (normal)

3 Request Bit. Same as arbitrary command bits, but usually used to indicate slave requests.

N0TE: Bits 7 to 3 may be cleared by the master by issuing the appropriate command bit (see above).

2 1 = Overrun detected (see discussion on overrun)

1 Read Hold. The NET/82 is in a wait state, waiting for the master to write to it.

0 Write Hold. The NET/82 is in a wait state, waiting for the master to read from it.

Bits seven through two are discussed elsewhere. The two hold status signals indicate that the slave has stopped, waiting for the master to read or write a byte to the data port. Normally this occurs just prior to and during an actual data transfer operation. However, the NET/82 may be left in this state for an indefinite period of time, since memory refreshing will occur. When in hold, no other operations occur in the NET/82, such as interrupt servicing or other processing.

SECTION 3.8.3 - OVERRUN

Overrun status is set on a NET/82 board whenever the master processor accesses the NET/82 data port (even) and the slave is not in a hold state (read or write). Since this is the essence of no-wait protocol, overrun status must be ignored when no-wait protocol is being used. Alternatively, the slave could use the overrun indication as part of the handshaking which shows that the master had in fact accessed the data port.

In normal operation, overrun is a fatal error condition, as the normal protocol specifies that the slave must be in the hold condition when the master is sending or receiving a data byte. For programming with 4MHz Z80A processors, the slave will usually do an INIR or OTIR instruction (block I/O), while the master will use a short loop consisting of NOP, OTI (or INI), and JNZ back. The extra NOP instruction and jump time is enough to ensure that the slave must wait for the master, no matter the slight variation in processor clock speeds. For other types and speeds of master processors, the protocol may be determined by calculation or empirically, as long as some slop is allowed for variations in individual processor boards.

While the overrun status bit may be read by either the master or the slave, only the slave may clear it, unless the master performs a total reset of the slave. Conversely, only the master, may set overrun, by accessing the data port as described above. Note that while the logic does detect timing problems, protocol problems must be detected by the master processor status read operation. The overrun logic does not get set if the master and slave are either both reading and both writing. Therefore the master should check for the type of hold (read or write) as part of the beginning of any slave I/O operating.

SECTION 3.8.4 – PROTOCOLS

Protocols are the province of the system software designer, and any discussion of this subject should be taken as guidance, rather than hard rules. Therefore, let us examine a typical protocol.

For a discussion of the typical protocol, we must abandon the master/slave vocabulary associated with the S-100 interface, and adopt the requestor/server nomenclature mo-re typical in a network environment. In this context, the NET/82 normally operates as requestor and the S-100 bus master normally operates as server. To initiate a request, the NET/82 would set the request bit in the local command port. He could then go on executing, servicing interrupts, or whatever, until the request was ready to be processed. This could be as simple as polling the request bit, waiting for it to go away in response to an external clear command, or it could involve an interrupt service routine of some type. However, nothing prevents the NET/82 from going directly to read or write hold, as necessary. Once the request is recognized, the NET/82 will go into write hold to transfer some of request descriptor block describing the transaction which is desired. This will then be followed, at an appropriate interval, by a read or write of the data, if necessary. Again, interrupts may optionally be used to initiate the second portion of the data transfer operation, especially if a disk operation must take place. Finally, both sides would check status and initiate error recovery, if appropriate.

One word of caution. If a server handles more than one requestor, it should be programmed such that no-erroneous operation of one requestor will crash the system. This implies timeout checks on any loop which waits on the NET/82.

NULL MODEM PADDLEBOARD

The MuSYS Corp. Null Modem Paddleboard is designed to interface the NET-81/82 family of computer boards to a terminal or printer via a standard RS-232-C serial communications link. When used, the system becomes an effective "DCE" device, and may communicate directly with any standard "DTE" device.

The RS-232-C lines recognized are:

1	FG	Frame Ground (optional)
2	TxD	Transmitted Data .
3	RxD	Received Data
5	CTS	Clear To Send
6	DSR	Data Set Ready
7	SG	Signal Ground
8	DCD	Data Carrier Detect
11	—	(Optional DTR)
14	—	(Optional DTR)
18	—	Reset Output (non-standard signal)
19	—	(Optional DTR, optional Reset Input)
20	DTR	Data Terminal Ready (optional)

There are three jumper areas used to select the above listed options. Looking at the component side of the board with the DB25 connector up: JA-1 is the two pads in the upper left, this area is not normally equipped with jumper pins; JA-2 is the two-pin jumper area in the lower right; and JA-3 is six-pin jumper area in the upper right. JA-3 is depicted in Figure 1. The options involved are:

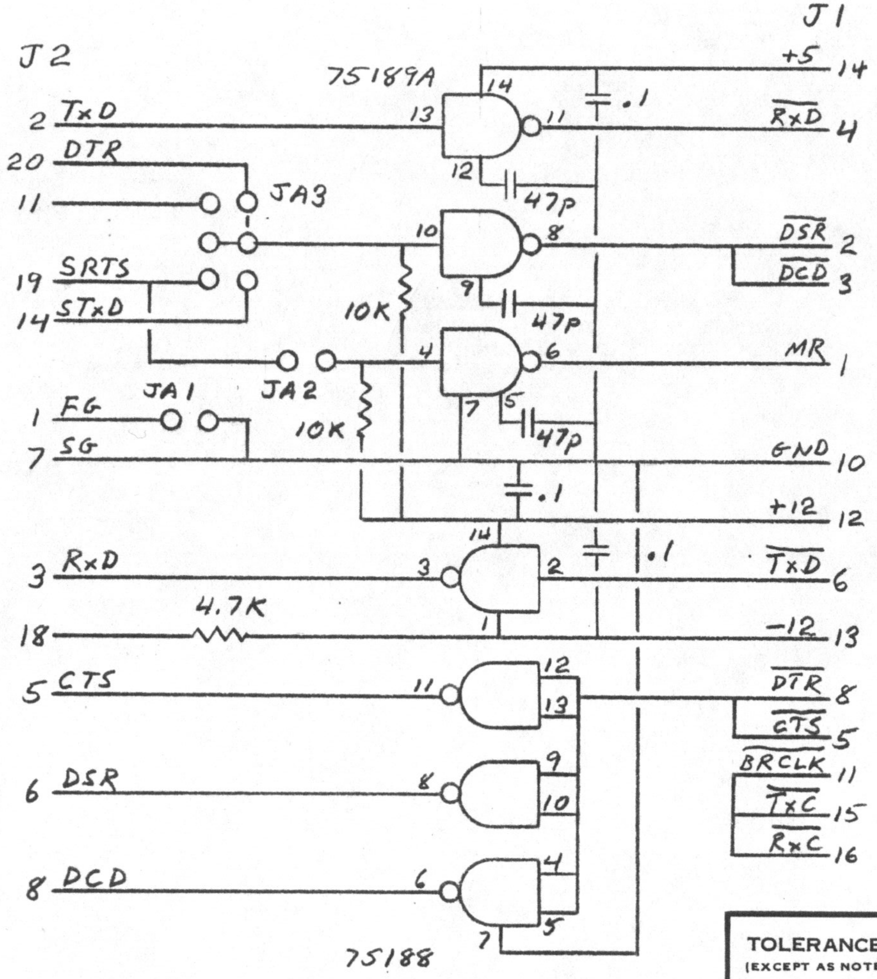
JA—1: When open, pin 1 (FG) is floating.
When jumped, pin 1 (FG) connects to ground.

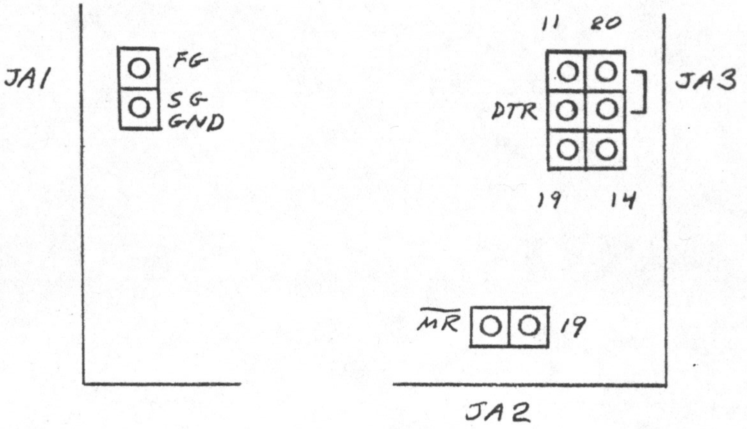
JA-2 When open, remote manual reset is disabled.
When jumped, shorting pin 18 to pin 19 causes reset. When this option is used, pin 19 may not be DTR.

JA-3 (See figure 1)
When open, DTR is always active.
When B-D jumped, DTR is pin 20 (normal).
When C-E jumped, DTR is pin 19 (NEC Spinwriter types).
When D-F jumped, DTR is pin 14 (Lear Slegler types).
When A-C jumped, DTR is pin 11 (TI-810 types).

```
    ---------
 11 | A | B | 20
    |---|---|
COM | C | D | COM         FIGURE 1
    |--- ---|
 19 | E | F | 14
    ---------
```

DATE	SYM	REVISION RECORD	AUTH.	DR.	CK.





TOLERANCES
(EXCEPT AS NOTED)

DECIMAL
±

FRACTIONAL
±

ANGULAR
±

SCALE

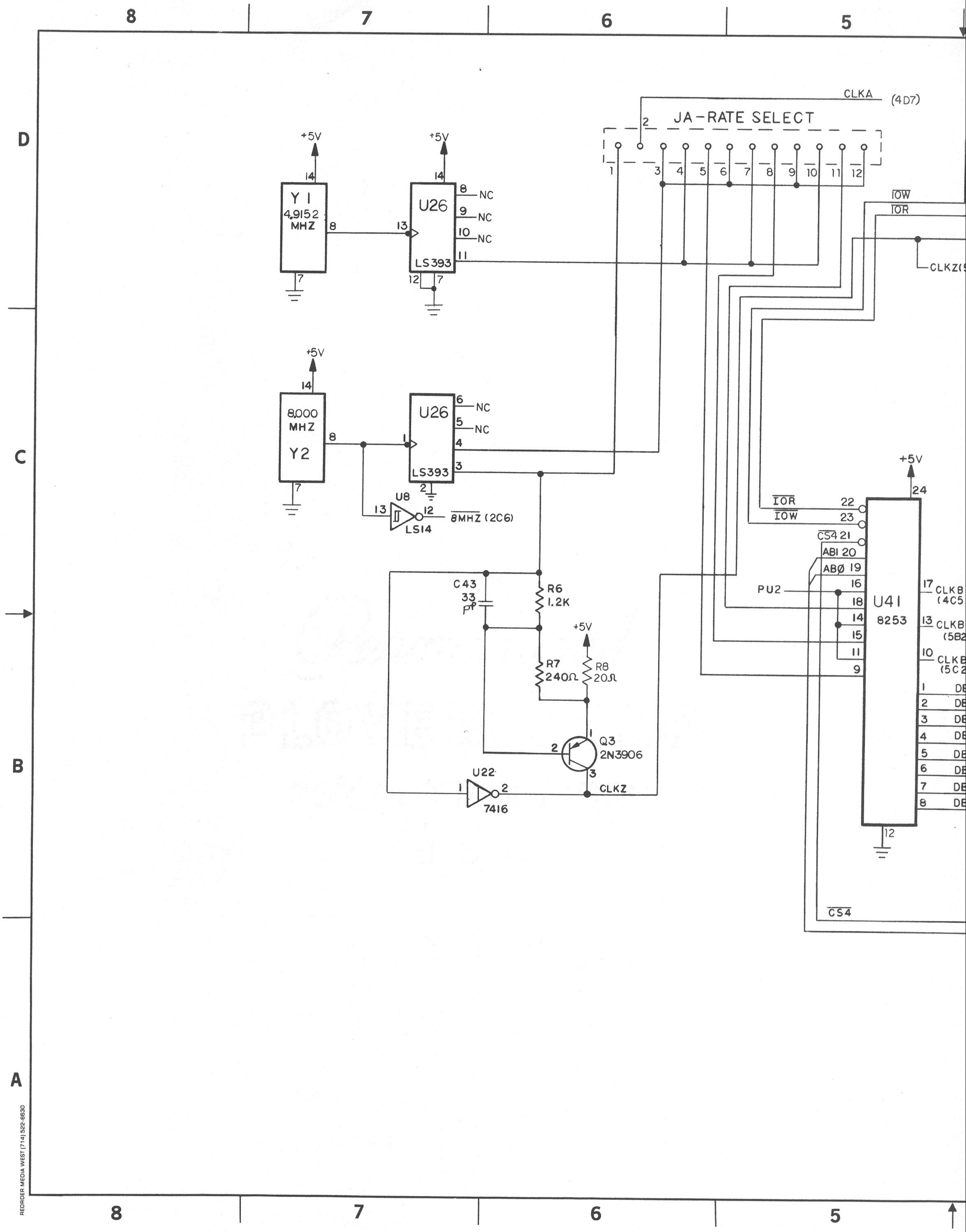
DRAWN BY

APPROVED BY

TITLE IM-1B
RS-232-C NULL MODEM INTERFACE

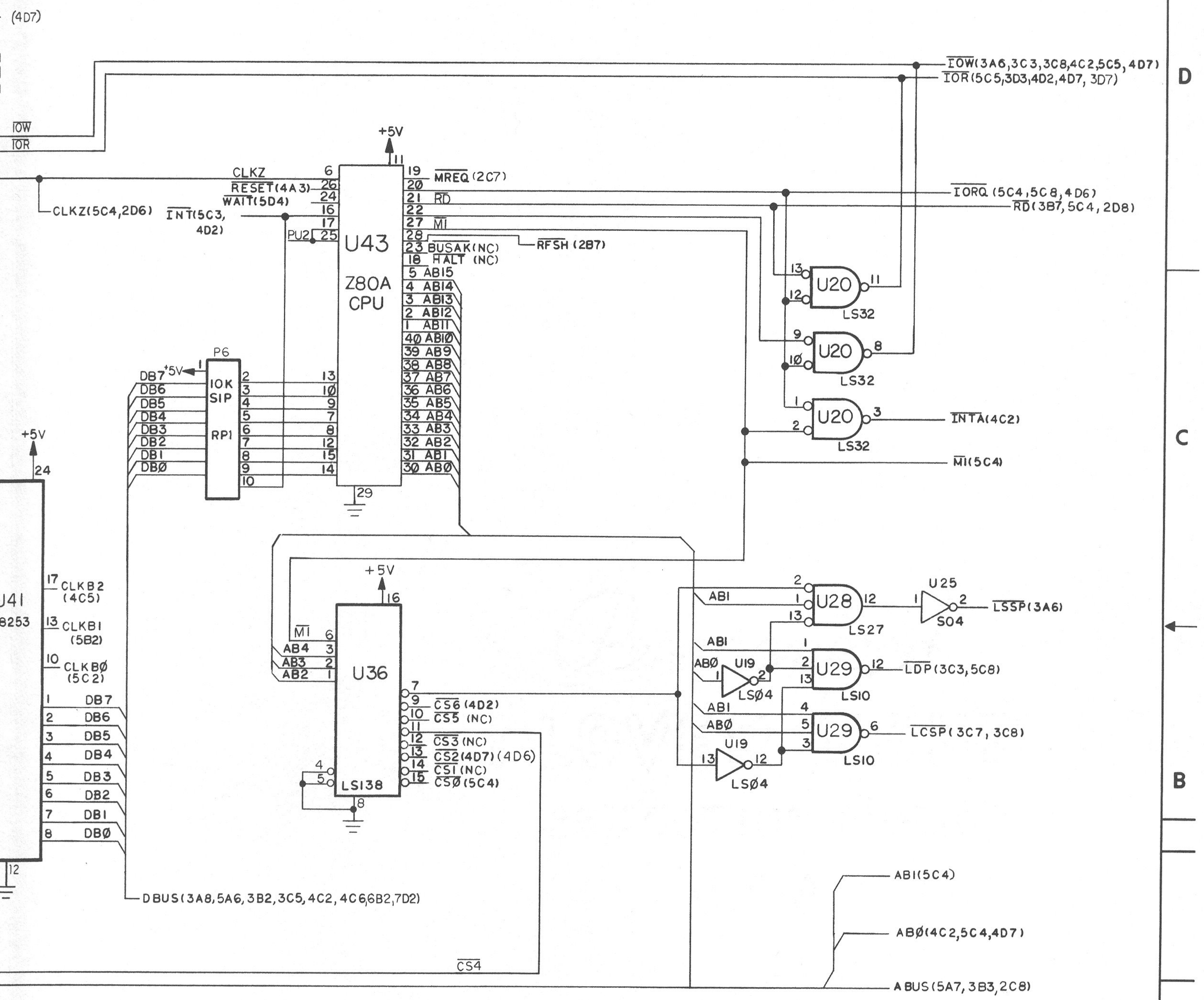
DATE
8-16-82

DRAWING NUMBER


8
7
6
5
D
C
B
A
CLKA
(4D7)
JA-RATE SELECT
1
2
3
4
5
6
7
8
9
10
11
12
+5V
14
Y1
4.9152
MHZ
8
7
U26
13
8
NC
9
NC
10
NC
LS393
11
12
IOW
IOR
CLKZ(5
+5V
14
8.000
MHZ
Y2
8
7
U26
1
6
NC
5
NC
4
LS393
3
2
U8
13
12
8MHZ (2C6)
LS14
C43
33
PF
R6
1.2K
+5V
R7
240Ω
R8
20Ω
1
Q3
2N3906
2
3
U22
1
2
7416
CLKZ
+5V
24
IOR
22
IOW
23
CS4 21
ABI 20
ABØ 19
16
PU2
18
14
15
11
9
U41
8253
17
CLKB
(4C5
13
CLKB
(5B2
10
CLKB
(5C2
1
DB
2
DB
3
DB
4
DB
5
DB
6
DB
7
DB
8
DB
12
CS4

4 3 2 1

REVISIONS				
ZONE	LTR	DESCRIPTION	DATE	APPROVED



QTY REQD	CODE IDENT	PART OR IDENTIFYING NO.	NOMENCLATURE OR DESCRIPTION	

PARTS LIST

UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES TOLERANCES ARE:
FRACTIONS ± DECIMALS .XX ± .XXX ± ANGLES ±

MATERIAL

FINISH

DO NOT SCALE DRAWING

CONTRACT NO.

APPROVALS		DATE
DRAWN	ASG DESIGNS	3-9-82
CHECKED		

MUSYS CORP
Specialists in Multi-user Microsystems
1451 E. Irvine Blvd., Suite 11, Tustin, CA 92680
(714) 730-5692 TWX: 910-595-1967
CABLE: MUSYSTSTN

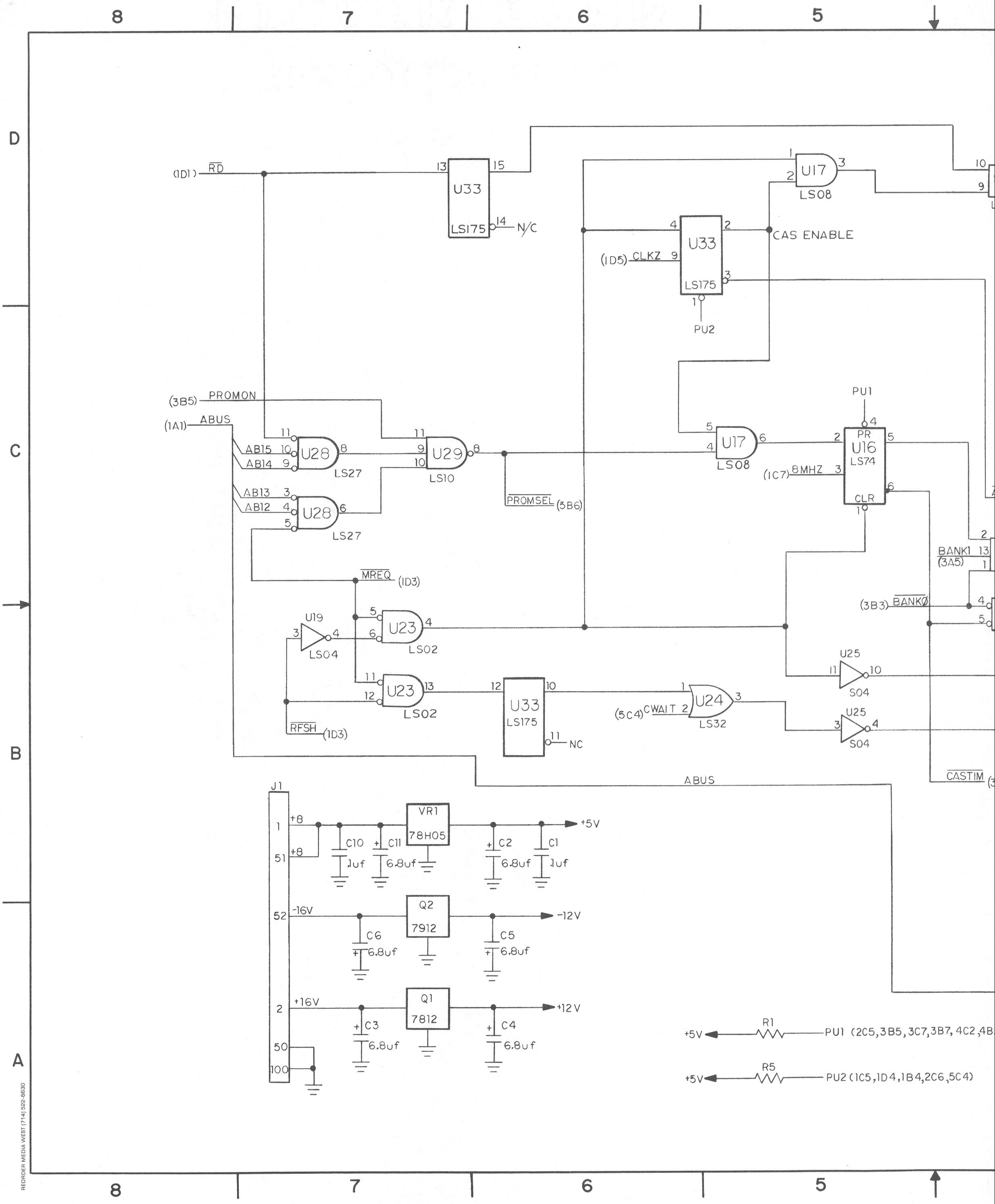
NEXT ASSY	USED ON
APPLICATION	

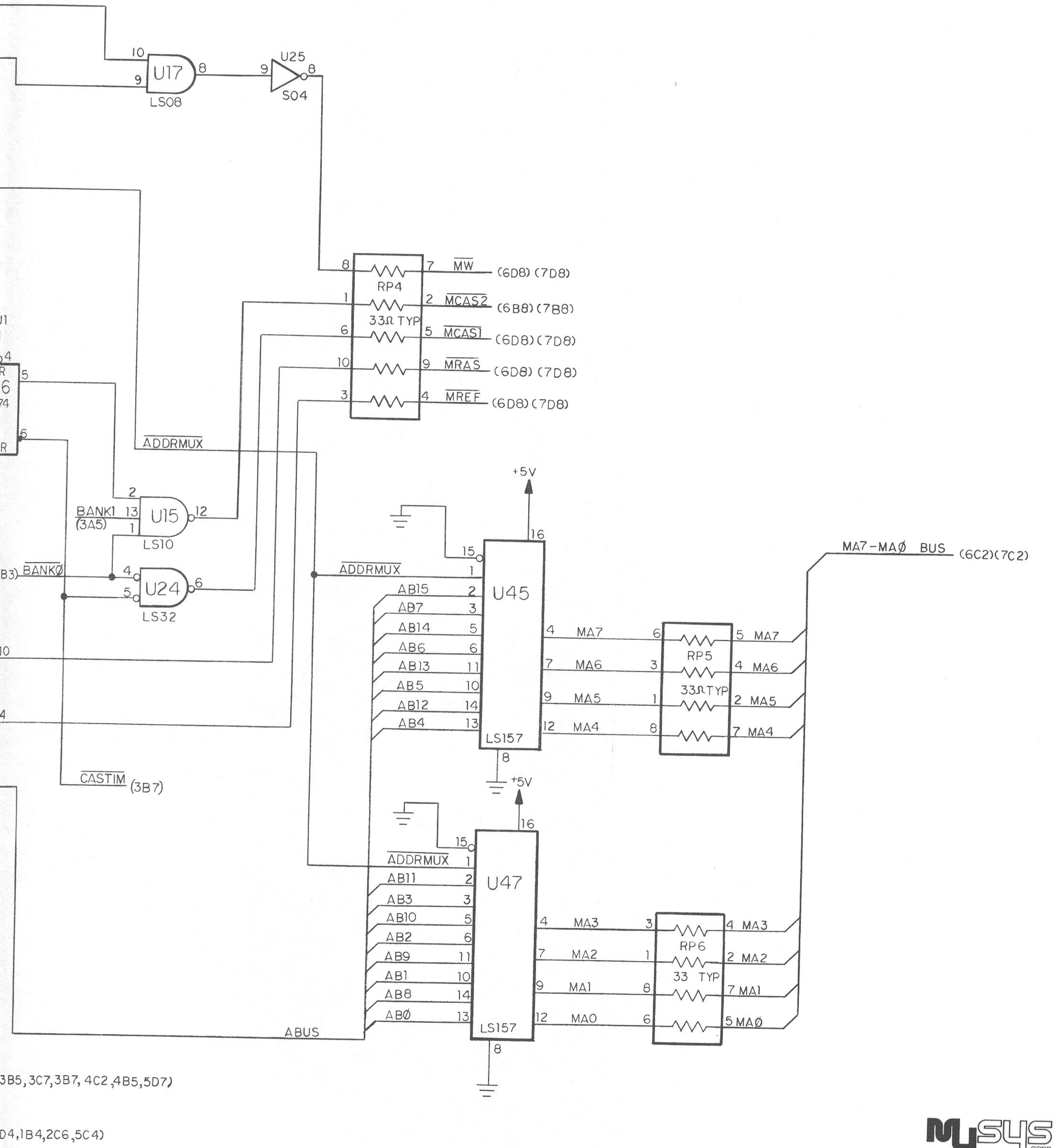
SIZE	CODE IDENT NO.	DRAWING NO.	REV
D			A

SCALE SHEET 1 OF 7

A

4 3 2 1

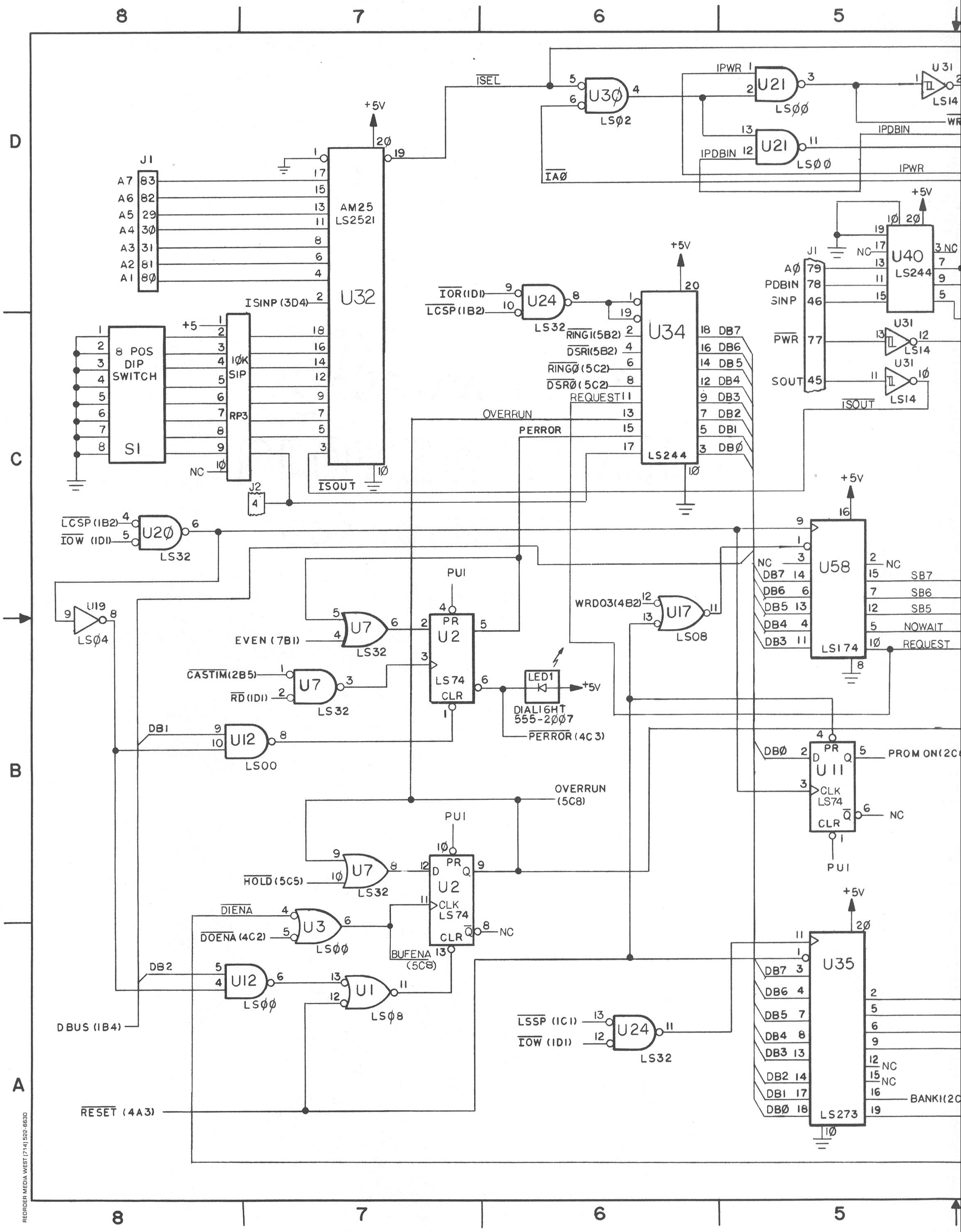

8
7
6
5
D
C
B
A
(1D1) RD
13 U33 LS175 15
14 N/C
1 2 U17 LS08 3
10
9
4 U33 LS175 2
CAS ENABLE
(1D5) CLKZ 9
3
1 PU2
PU1
4 PR
5 U17 LS08 6
2 U16 LS74 5
(1C7) 8MHZ 3
CLR 1 6
(3B5) PROMON
(1A1) ABUS
AB15 10
AB14 9
11 U28 8 LS27
11 9 10 U29 8 LS10
PROMSEL (3B6)
AB13 3
AB12 4
5 U28 6 LS27
MREQ (1D3)
U19 3 LS04 4
5 6 U23 4 LS02
11 12 U23 13 LS02
RFSH (1D3)
12 U33 LS175 10
11 NC
(5C4) CWAIT 2
1 U24 3 LS32
2
BANK1 13
(3A5) 1
(3B3) BANKØ 4
5
U25 11 10 S04
U25 3 4 S04
CASTIM (3
ABUS
J1
1 +8
51 +8
VR1 78H05
C10 .1uf
C11 6.8uf
C2 6.8uf
C1 .1uf
+5V
52 -16V
C6 6.8uf
Q2 7912
C5 6.8uf
-12V
2 +16V
C3 6.8uf
Q1 7812
C4 6.8uf
+12V
50
100
+5V R1 PU1 (2C5,3B5,3C7,3B7,4C2,4B
+5V R5 PU2 (1C5,1D4,1B4,2C6,5C4)



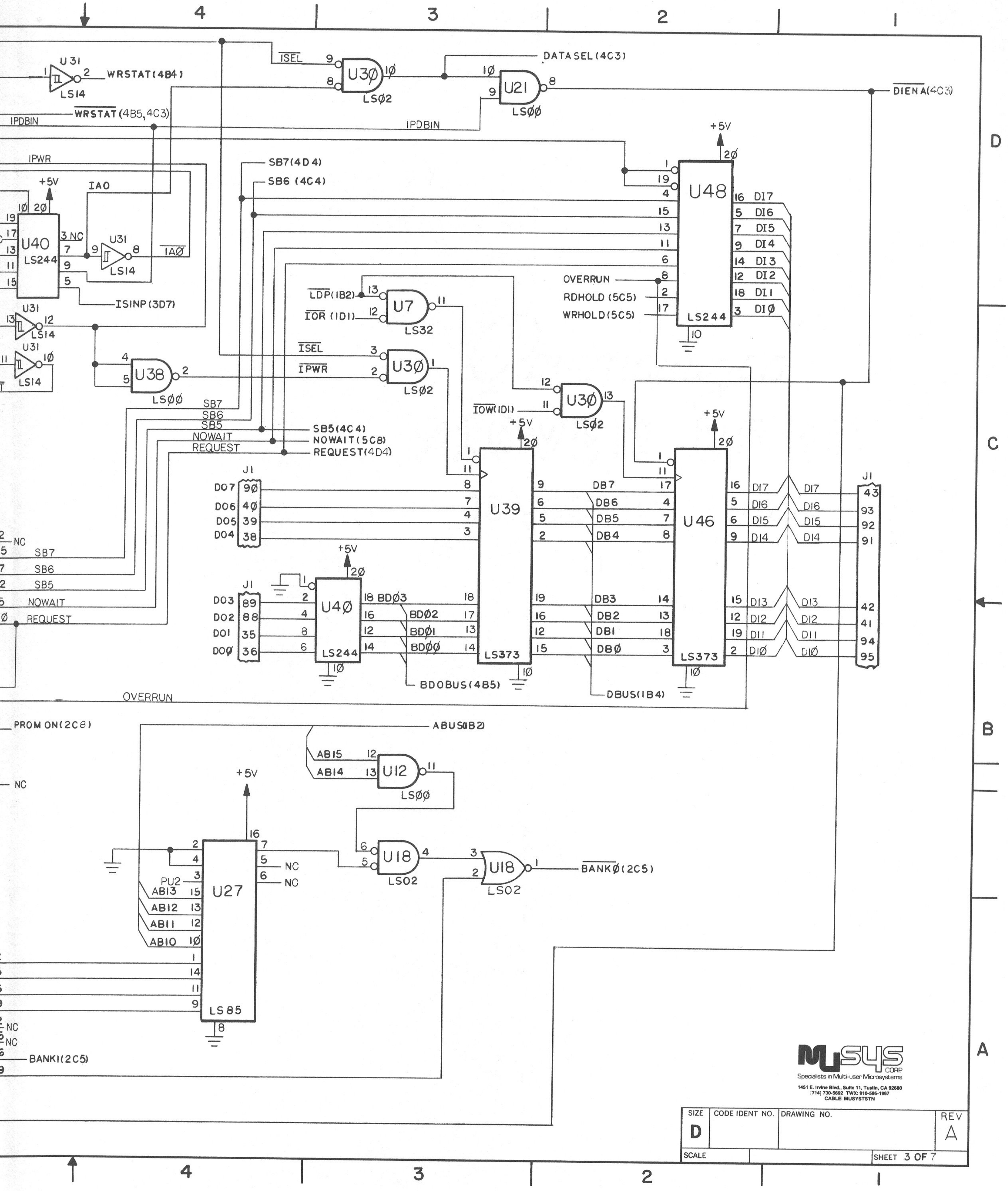
MUSYS
CORP
Specialists in Multi-user Microsystems

1451 E. Irvine Blvd., Suite 11, Tustin, CA 92680
(714) 730-5692 TWX: 910-595-1967
CABLE: MUSYSTSTN

SIZE	CODE IDENT NO.	DRAWING NO.	REV
D			A
SCALE		SHEET 2 OF 7	

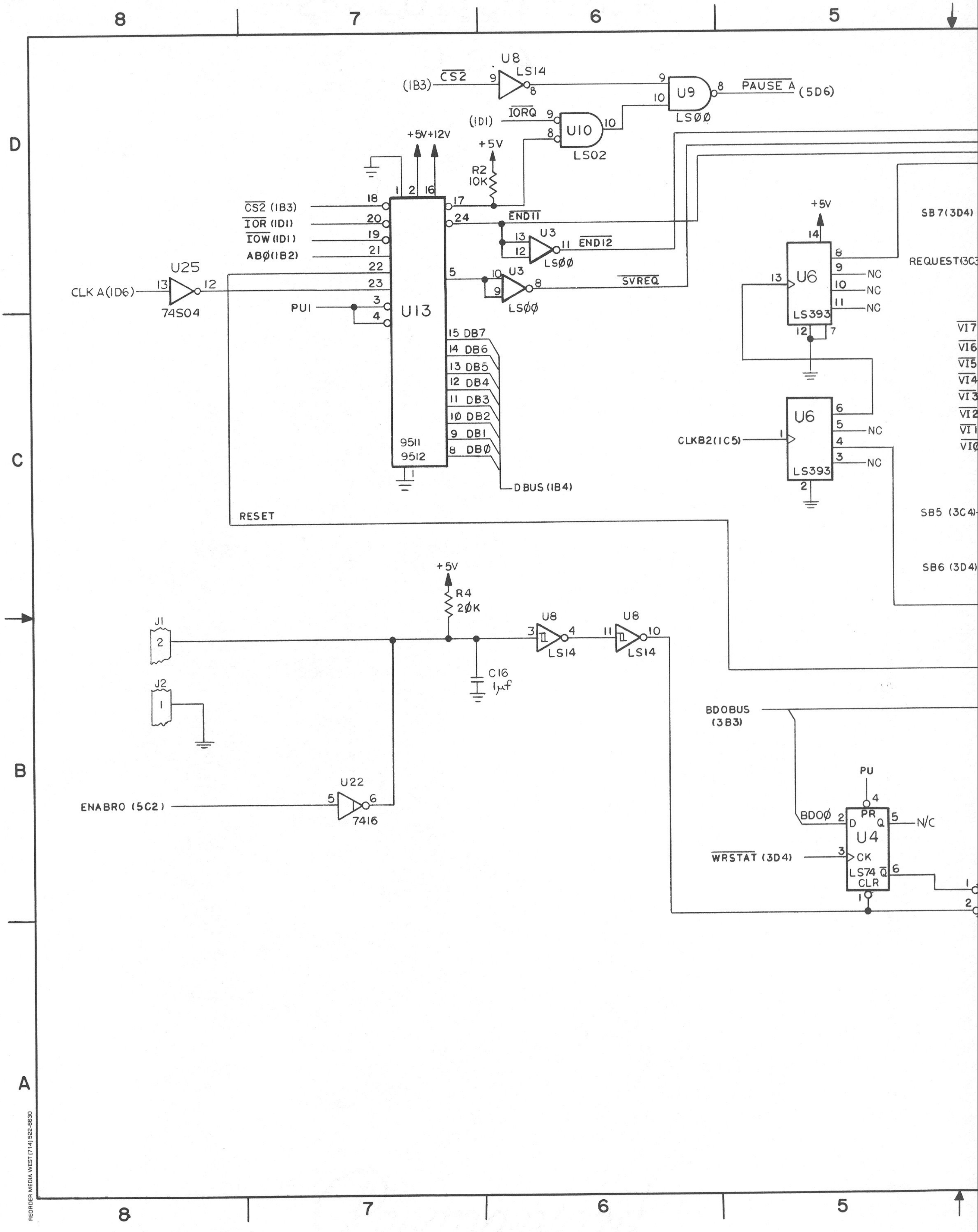


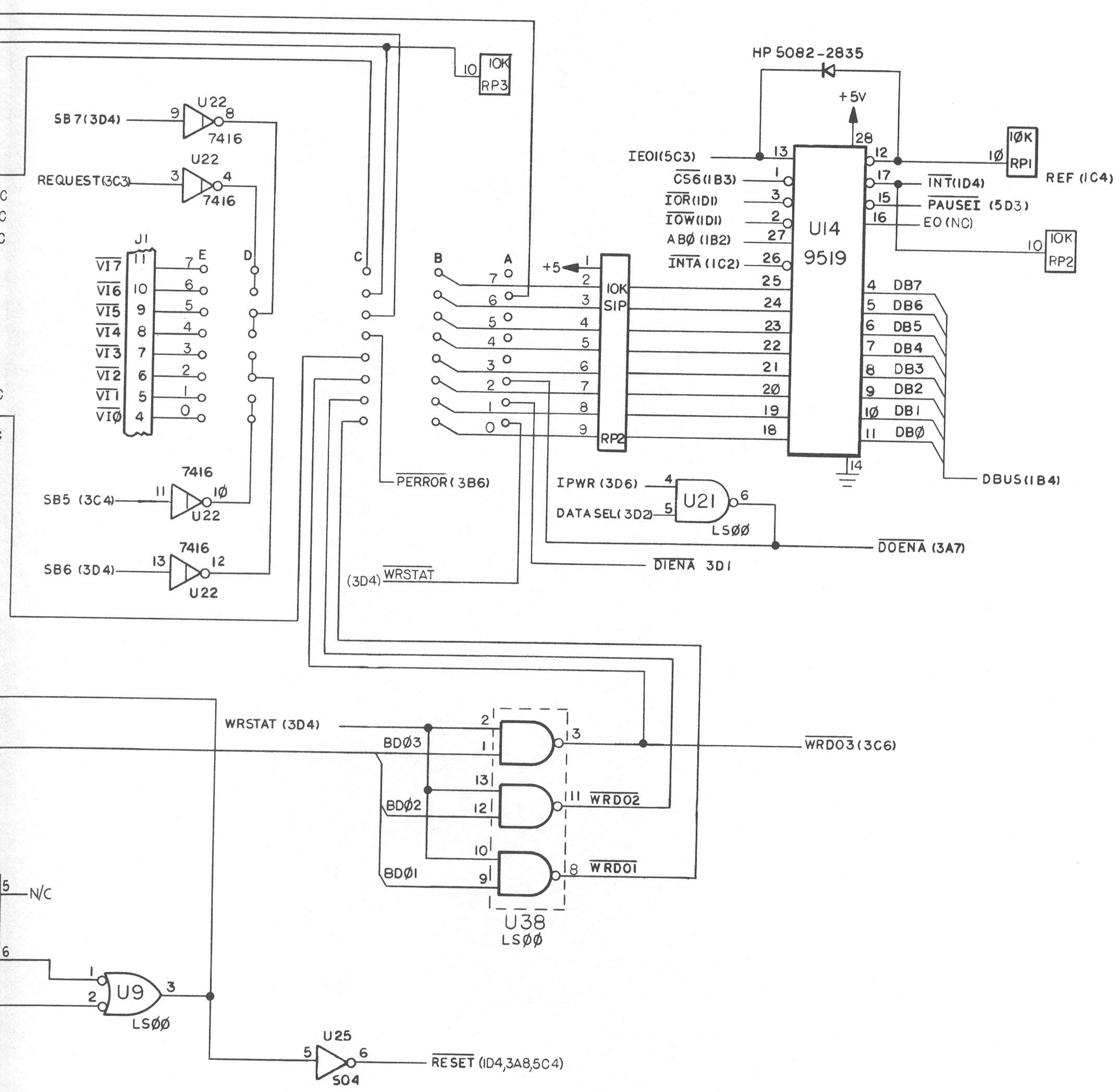
8
7
6
5
D
C
B
A
J1
A7 83
A6 82
A5 29
A4 3Ø
A3 31
A2 81
A1 8Ø
+5V
AM25 LS2521
U32
ISINP (3D4)
ISEL
ISOUT
8 POS DIP SWITCH
S1
+5
1ØK SIP
RP3
NC
J2
4
U3Ø
LSØ2
IAØ
IPWR
U21
LSØØ
IPDBIN
U31
LS14
WR
IPDBIN
IPWR
IOR (1D1)
LCSP (1B2)
U24
LS32
RING1 (5B2)
DSR1 (5B2)
RINGØ (5C2)
DSRØ (5C2)
REQUEST
OVERRUN
PERROR
U34
LS244
DB7
DB6
DB5
DB4
DB3
DB2
DB1
DBØ
U40
LS244
NC
AØ 79
PDBIN 78
SINP 46
PWR 77
SOUT 45
ISOUT
LCSP (1B2)
IOW (1D1)
U2Ø
LS32
U19
LSØ4
PU1
U7
LS32
EVEN (7B1)
U2
PR
LS74
CLR
CASTIM (2B5)
RD (1D1)
LED1
DIALIGHT 555-2ØØ7
PERROR (4C3)
DB1
U12
LS00
U58
LS174
WRDO3 (4B2)
U17
LS08
SB7
SB6
SB5
NOWAIT
REQUEST
DBØ
U11
CLK
PROM ON (2C
OVERRUN (5C8)
HOLD (5C5)
D
Q
DIENA
DOENA (4C2)
U3
LSØØ
BUFENA (5C8)
NC
DB2
U12
LSØØ
U1
LSØ8
DBUS (1B4)
RESET (4A3)
LSSP (1C1)
IOW (1D1)
U24
LS32
U35
LS273
BANK1 (2C



MUSYS CORP.
Specialists in Multi-user Microsystems
1451 E. Irvine Blvd., Suite 11, Tustin, CA 92680
(714) 730-5692 TWX: 910-595-1967
CABLE: MUSYSTSTN

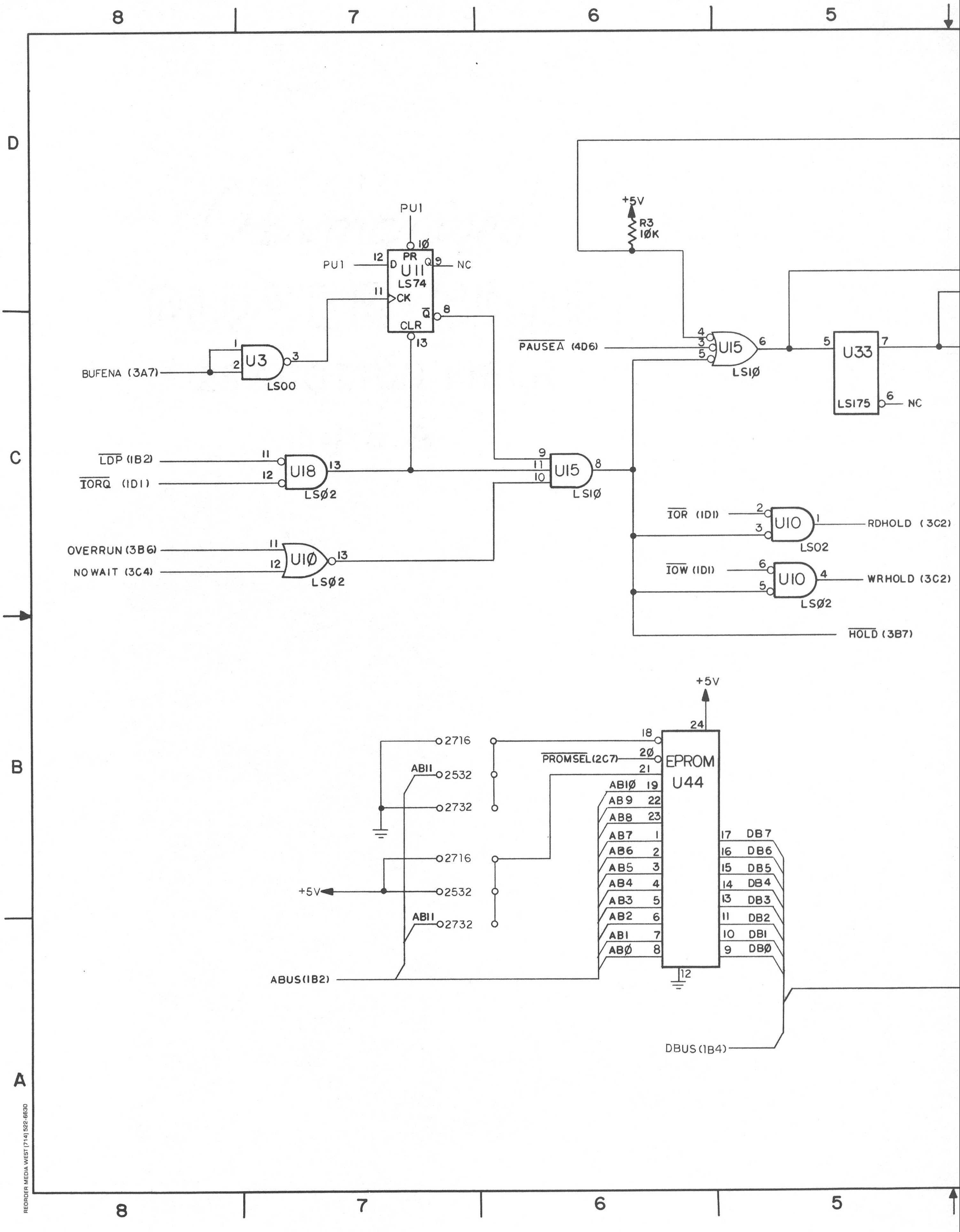
SIZE	CODE IDENT NO.	DRAWING NO.	REV
D			A
SCALE		SHEET 3 OF 7	

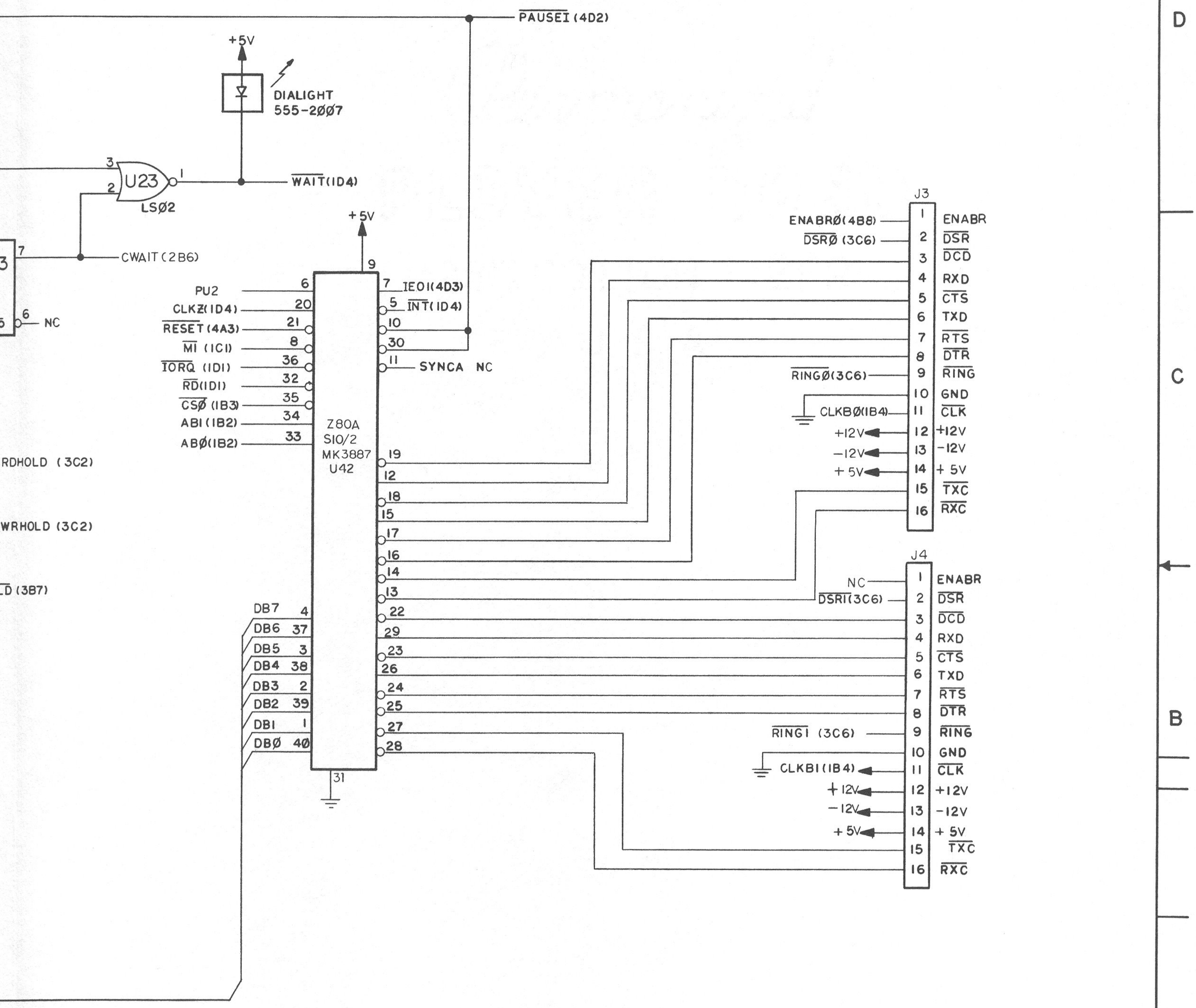

8
7
6
5
D
C
B
A
U8
LS14
(1B3)
CS2
9
8
U9
LS00
PAUSE A
(5D6)
10
(1D1)
IORQ
U10
LS02
+5V+12V
+5V
R2
10K
1
2
16
17
24
18
20
19
21
22
23
3
4
5
CS2 (1B3)
IOR (1D1)
IOW (1D1)
AB0(1B2)
END11
U3
13
12
11
END12
LS00
SVREQ
U25
CLK A(1D6)
74S04
PU1
U13
15 DB7
14 DB6
13 DB5
12 DB4
11 DB3
10 DB2
9 DB1
8 DB0
9511
9512
DBUS (1B4)
RESET
+5V
U6
LS393
NC
SB7(3D4)
REQUEST(3C3)
VI7
VI6
VI5
VI4
VI3
VI2
VI1
VI0
CLKB2(1C5)
SB5 (3C4)
SB6 (3D4)
+5V
R4
20K
J1
J2
C16
1μf
U8
LS14
BDOBUS
(3B3)
U22
ENABRO (5C2)
7416
PU
BD00
D
PR
Q
N/C
U4
WRSTAT (3D4)
CK
LS74 Q
CLR



MUSYS CORP.
Specialists in Multi-user Microsystems
1451 E. Irvine Blvd., Suite 11, Tustin, CA 92680
(714) 730-5692 TWX: 910-595-1967
CABLE: MUSYSTSTN

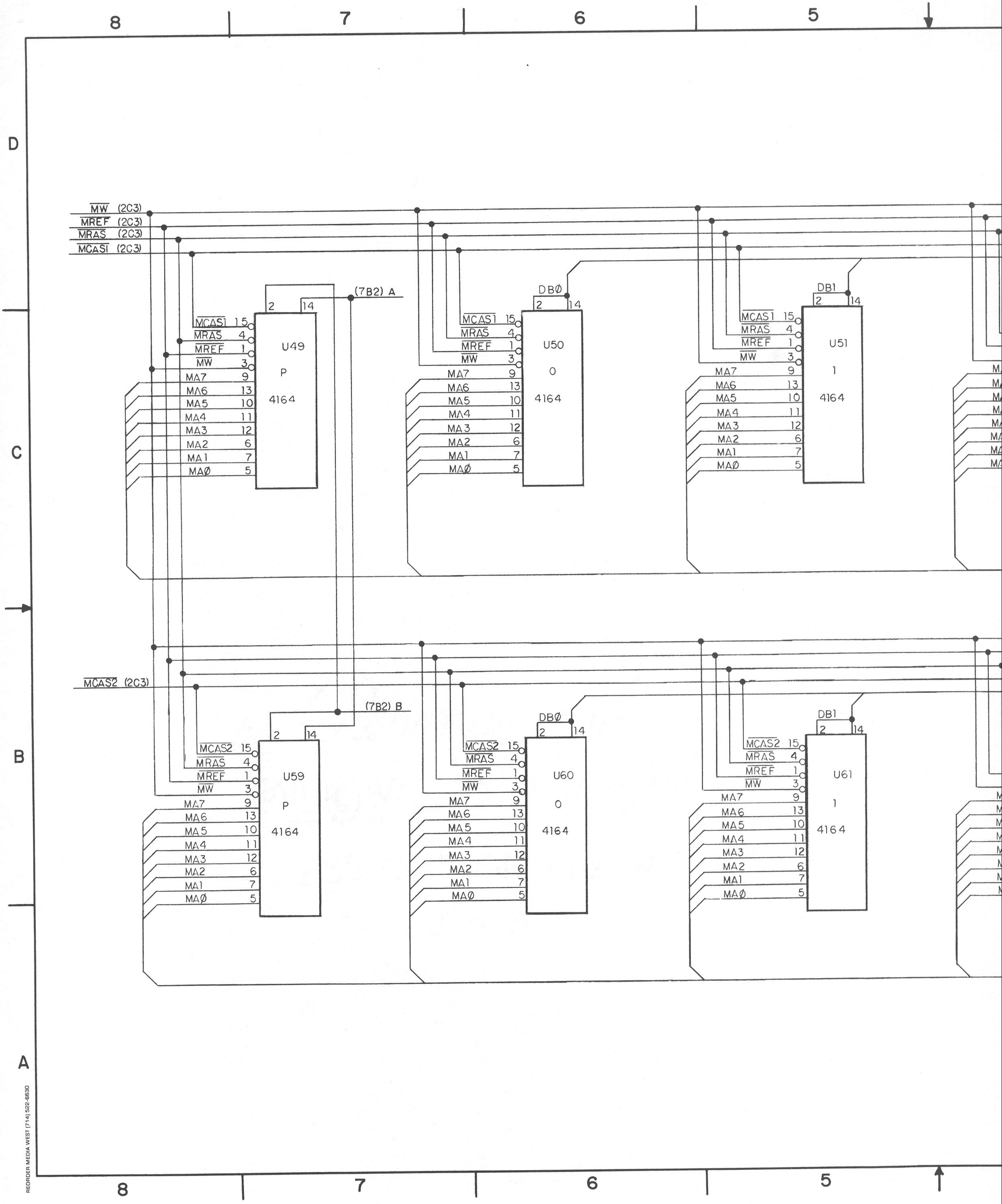
SIZE	CODE IDENT NO.	DRAWING NO.	REV
D			A
SCALE		SHEET 4 OF 7	


8
7
6
5
D
C
B
A
PU1
1Ø
PU1
12
D
PR
U11
Q
9
NC
LS74
11
CK
Q
8
CLR
13
1
2
U3
3
LS00
BUFENA (3A7)
+5V
R3
1ØK
4
3
5
U15
6
LS1Ø
PAUSEA (4D6)
5
U33
7
LS175
6
NC
LDP (1B2)
IORQ (1D1)
11
12
U18
13
LSØ2
9
11
10
U15
8
LS1Ø
IOR (1D1)
2
3
U10
1
LS02
RDHOLD (3C2)
IOW (1D1)
6
5
U10
4
LSØ2
WRHOLD (3C2)
HOLD (3B7)
OVERRUN (3B6)
NOWAIT (3C4)
11
12
U1Ø
13
LSØ2
+5V
24
18
PROMSEL(2C7)
20
21
EPROM
U44
2716
AB11
2532
2732
2716
+5V
2532
AB11
2732
ABUS(1B2)
AB1Ø 19
AB9 22
AB8 23
AB7 1
AB6 2
AB5 3
AB4 4
AB3 5
AB2 6
AB1 7
ABØ 8
17 DB7
16 DB6
15 DB5
14 DB4
13 DB3
11 DB2
10 DB1
9 DBØ
12
DBUS (1B4)
8
7
6
5

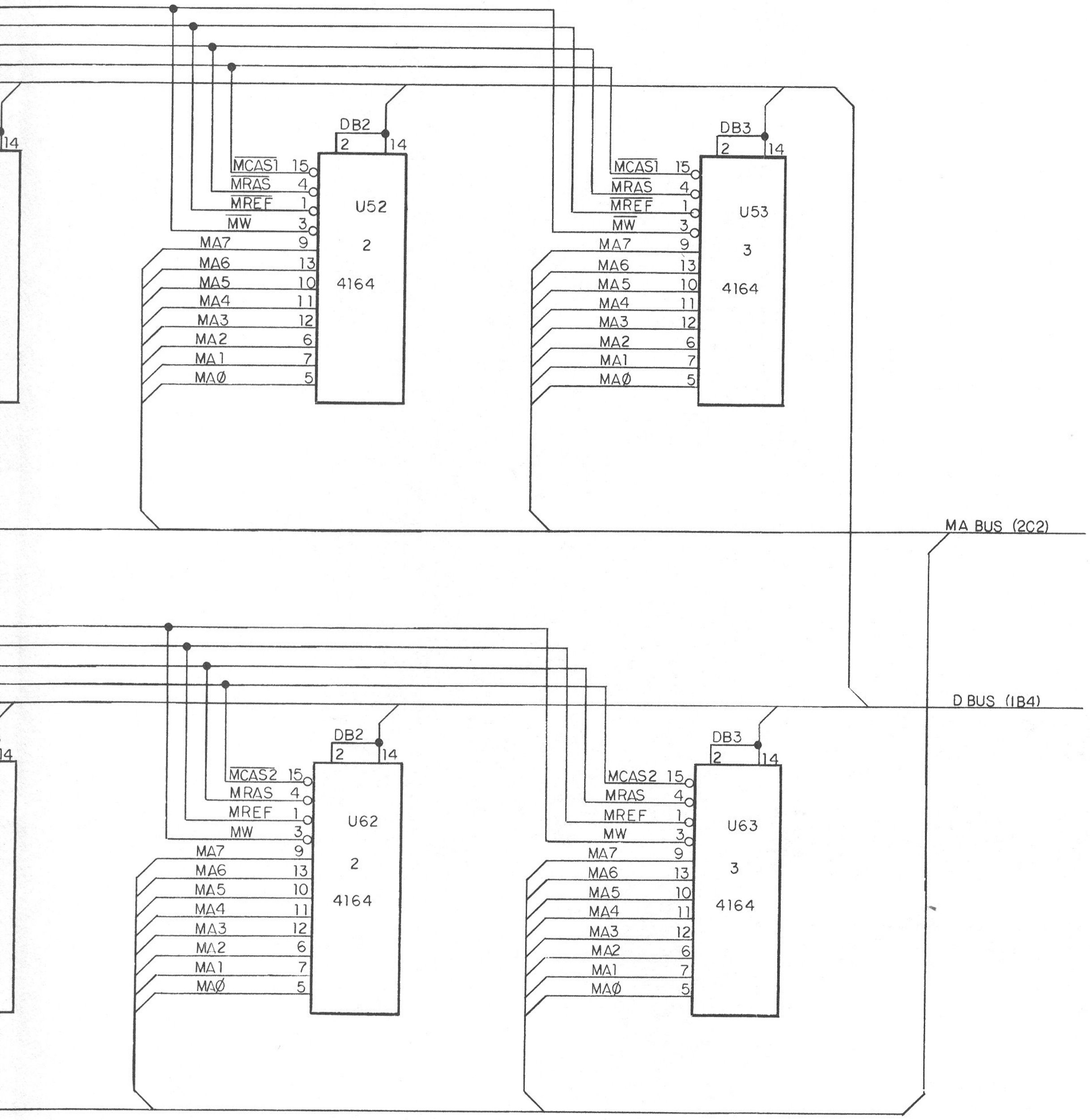


MUSYS CORP.
Specialists in Multi-user Microsystems
1451 E. Irvine Blvd., Suite 11, Tustin, CA 92680
[714] 730-5692 TWX: 910-595-1967
CABLE: MUSYSTSTN

SIZE	CODE IDENT NO.	DRAWING NO.	REV
D			A
SCALE		SHEET 5 OF 7	

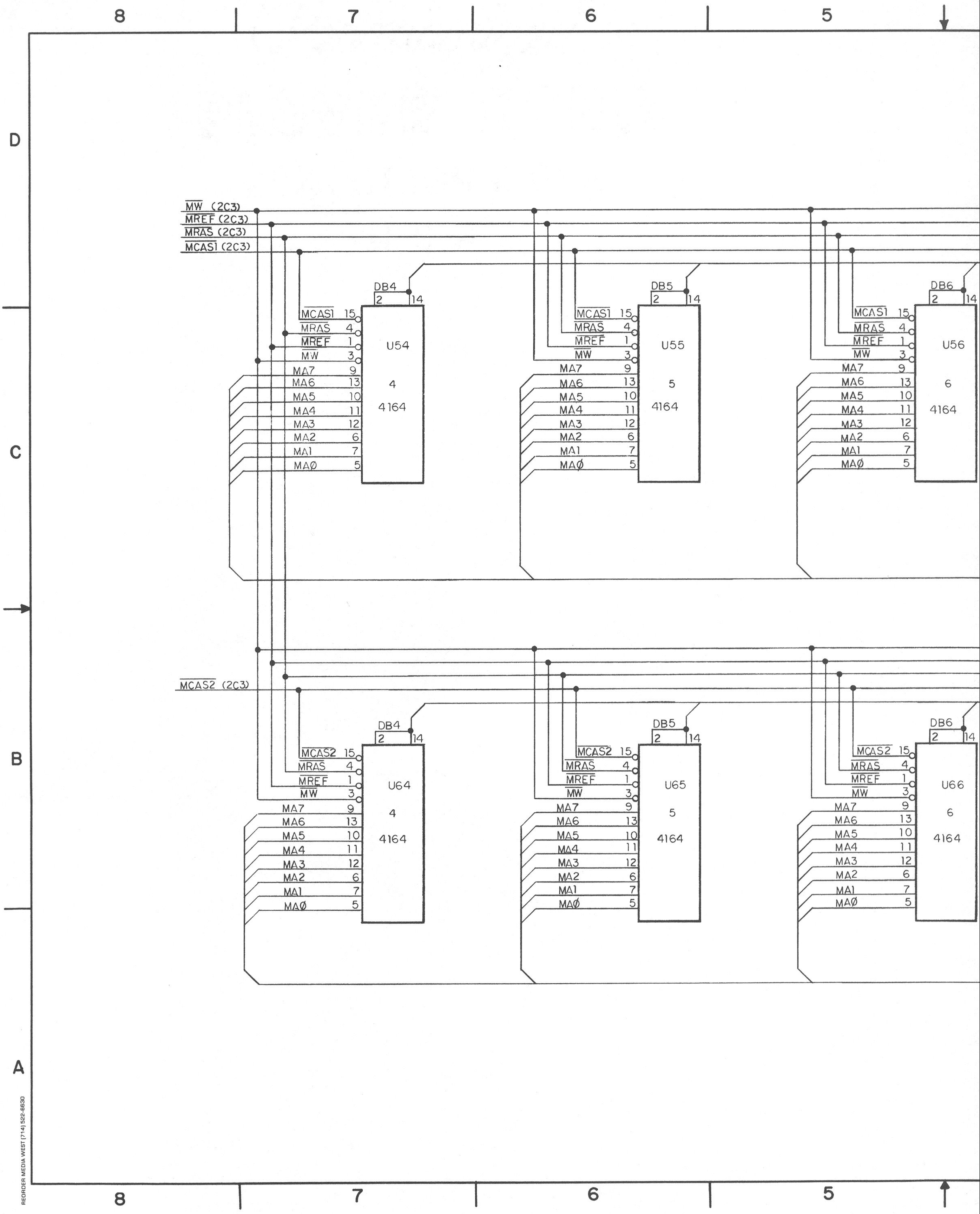


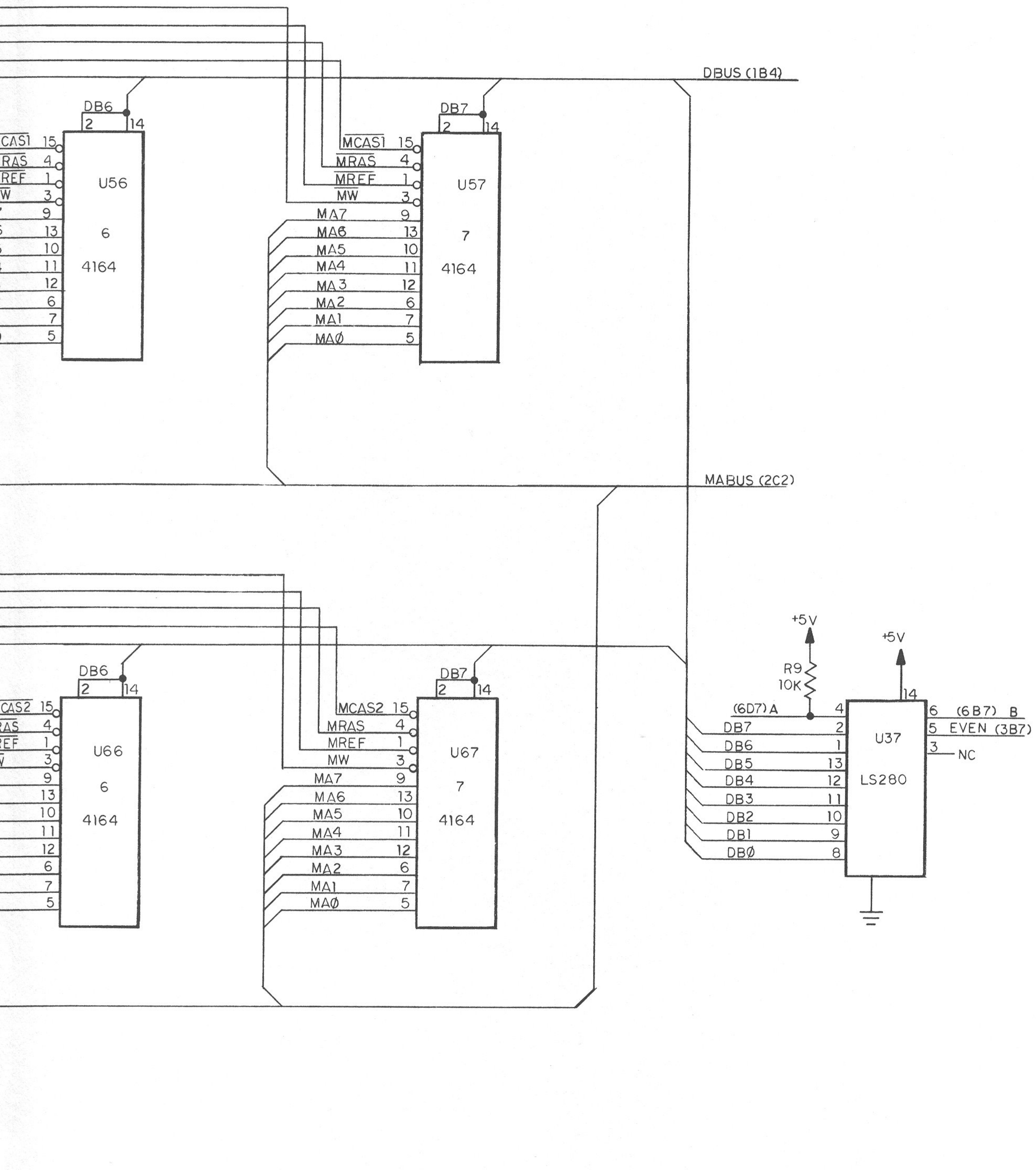
8
7
6
5
D
C
B
A
MW (2C3)
MREF (2C3)
MRAS (2C3)
MCAS1 (2C3)
(7B2) A
DBØ
DB1
U49
P
4164
U50
0
4164
U51
1
4164
MCAS1 15
MRAS 4
MREF 1
MW 3
MA7 9
MA6 13
MA5 10
MA4 11
MA3 12
MA2 6
MA1 7
MAØ 5
2
14
MCAS2 (2C3)
(7B2) B
U59
P
4164
U60
0
4164
U61
1
4164
MCAS2 15
MRAS 4
MREF 1
MW 3
MA7 9
MA6 13
MA5 10
MA4 11
MA3 12
MA2 6
MA1 7
MAØ 5



MUSYS CORP
Specialists in Multi-user Microsystems
1451 E. Irvine Blvd., Suite 11, Tustin, CA 92680
(714) 730-5692 TWX: 910-595-1967
CABLE: MUSYSTSTN

SIZE	CODE IDENT NO.	DRAWING NO.	REV
D			A
SCALE		SHEET 6 OF 7	


8
7
6
5
D
C
B
A
MW (2C3)
MREF (2C3)
MRAS (2C3)
MCAS1 (2C3)
MCAS2 (2C3)
DB4
DB5
DB6
2
14
MCAS1 15
MCAS2 15
MRAS 4
MREF 1
MW 3
MA7 9
MA6 13
MA5 10
MA4 11
MA3 12
MA2 6
MA1 7
MAØ 5
U54
4
4164
U55
5
4164
U56
6
4164
U64
4
4164
U65
5
4164
U66
6
4164



MuSYS CORP.

Specialists in Multi-user Microsystems

1451 E. Irvine Blvd., Suite 11, Tustin, CA 92680
[714] 730-5692 TWX: 910-595-1967
CABLE: MUSYSTSTN

SIZE D	CODE IDENT NO.	DRAWING NO.	REV A
SCALE		SHEET 7 OF 7	