Racal 6065P-4

4-Channel Serial Interface Card with Programmable Data Rate

6065

VXI Serial Interface Module

RACAL INSTRUMENTS

RACAL
The Electronics Group

PUBLICATION NO. 980728

6065

VXI Serial Interface Module

RACAL INSTRUMENTS

RACAL INSTRUMENTS INC.
4 Goodyear Street, P.O. Box C-19541, Irvine, CA 92718
Telephone: 714/859-8999, Fax: 714/859-2505

RACAL INSTRUMENTS, LTD.
480 Bath Road, Slough, Berkshire, SL1 6BE, United Kingdom
Telephone: (0)628 604455, Fax: (0)628 662017

RACAL SYSTEMS ELECTRONIQUE S.A.
18 Avenue Dutartre, 78150 Le Chesnay, France
Telephone: (1)3955 8888, Fax: (1)3955 6735

RACAL ELEKTRONIK SYSTEMS GmbH
Frankenforster Strasse 21, 51427 Bergisch Gladbach, Germany
Telephone: 02204 92220, Fax: 02204 21491

RACAL SYSTEMS ELETTRONICA srl
Strada 2 - Palazzo C 4, 20090 Milanofiori Assago, Milano, Italy
Telephone: (02) 5750.1796, Fax: (02) 5750.1828

RACAL

PUBLICATION DATE: SEPTEMBER 12, 1994

WARRANTY STATEMENT

Products sold by Racal Instruments Inc. are warranted to be free from defects in workmanship or materials. Racal Instruments Inc. will, at its option, either repair or replace any hardware products which prove to be defective during the warranty period. You are a valued customer. Our mission is to make any necessary repairs in a reliable and timely manner.

Duration of Warranty

The warranty period for all Racal Instruments Inc. hardware is one year, except software and firmware products designated for use with Racal Instruments Inc. hardware are warranted not to fail to execute their programming instructions due to defect in materials or workmanship for a period of ninety (90) days from the date of delivery to the initial end user.

Return of Product

Authorization is required from Racal Instruments before you send us your product. Call Customer Support at 1-800-722-3262. You will be given repacking instructions. You will be required to return your product prepaid UPS. Racal Instruments will pay the shipping costs back to you UPS Ground.

Limitation of Warranty

Racal Instruments Inc. shall be released from all obligations under this warranty in the event repairs or modifications are made by persons other than authorized Racal Instruments Inc. service personnel or without the written consent of Racal Instruments Inc.

Racal Instruments Inc. expressly disclaims any liability to its customers, dealers and representatives and to users of its products, and to any other person or persons, for special or consequential damages of any kind and from any cause whatsoever arising out of or in any way connected with the manufacture, sale, handling, repair, maintenance, replacement or use of said products.

Representations and warranties made by any person, including dealers and representatives of Racal Instruments Inc., which are inconsistent or in conflict with the terms of this warranty (including but not limited to the limitations of the liability of Racal Instruments Inc. as set forth above), shall not be binding upon Racal Instruments Inc. unless reduced to writing and approved by an officer of Racal Instruments Inc.

Except as stated above, Racal Instruments Inc. makes no warranty, express or implied (either in fact or by operation of law), statutory or otherwise; and except to the extent stated above, Racal Instruments Inc. shall have no liability under any warranty, express or implied (either in fact or by operation of law), statutory or otherwise.

PROPRIETARY NOTICE

This document and the technical data herein disclosed, are proprietary to Racal Instruments, and shall not, without express written permission of Racal Instruments, be used, in whole or in part to solicit quotations from a competitive source or used for manufacture by anyone other than Racal Instruments. The information herein has been developed at private expense, and may only be used for operation and maintenance reference purposes or for purposes of engineering evaluation and incorporation into technical specifications and other documents which specify procurement of products from Racal Instruments.

FOR YOUR SAFETY

Before undertaking any troubleshooting, maintenance or exploratory procedure, read carefully the **WARNING** and **CAUTION** notices.

This equipment contains voltage hazardous to human life and safety, and is capable of inflicting personal injury.

If this instrument is to be powered from the AC line (mains) through an autotransformer, ensure the common connector is connected to the neutral (earthed pole) of the power supply.

Before operating the unit, ensure the conductor (green wire) is connected to the ground (earth) conductor of the power outlet. Do not use a two-conductor extension cord or a three-prong/two-prong adaptor. This will defeat the protective feature of the third conductor in the power cord.

Maintenance and calibration procedures sometimes call for operation of the unit with power applied and protective covers removed. Read the procedures and heed warnings to avoid "live" circuit points.

Before operating this instrument:

1. Ensure the instrument is configured to operate on the voltage at the power source. See Installation Section.

2. Ensure the proper fuse is in place for the power source to operate.

3. Ensure all other devices connected to or in proximity to this instrument are properly grounded or connected to the protective third-wire earth ground.

If the instrument:

- fails to operate satisfactorily
- shows visible damage
- has been stored under unfavorable conditions
- has sustained stress

Do not operate until performance is checked by qualified personnel.

TABLE OF CONTENTS

MODEL 6065

SECTION 1 GENERAL INFORMATION

SECTION 2 INSTALLATION

SECTION 3 OPERATION

SECTION 4 THEORY OF OPERATION

TABLE OF CONTENTS (continued)

FIGURES

1.1 INTRODUCTION

This manual provides the necessary information to install the Racal Instruments Model 6065 Serial Interface Module in a VXIbus compatible chassis, and to correctly operate the module.

1.2 GENERAL INFORMATION

The Model 6065 Serial Interface module is a C-size VXIbus compatible card. The module is available with four or eight channels. Each channel can be configured as one of four standard interface buses.

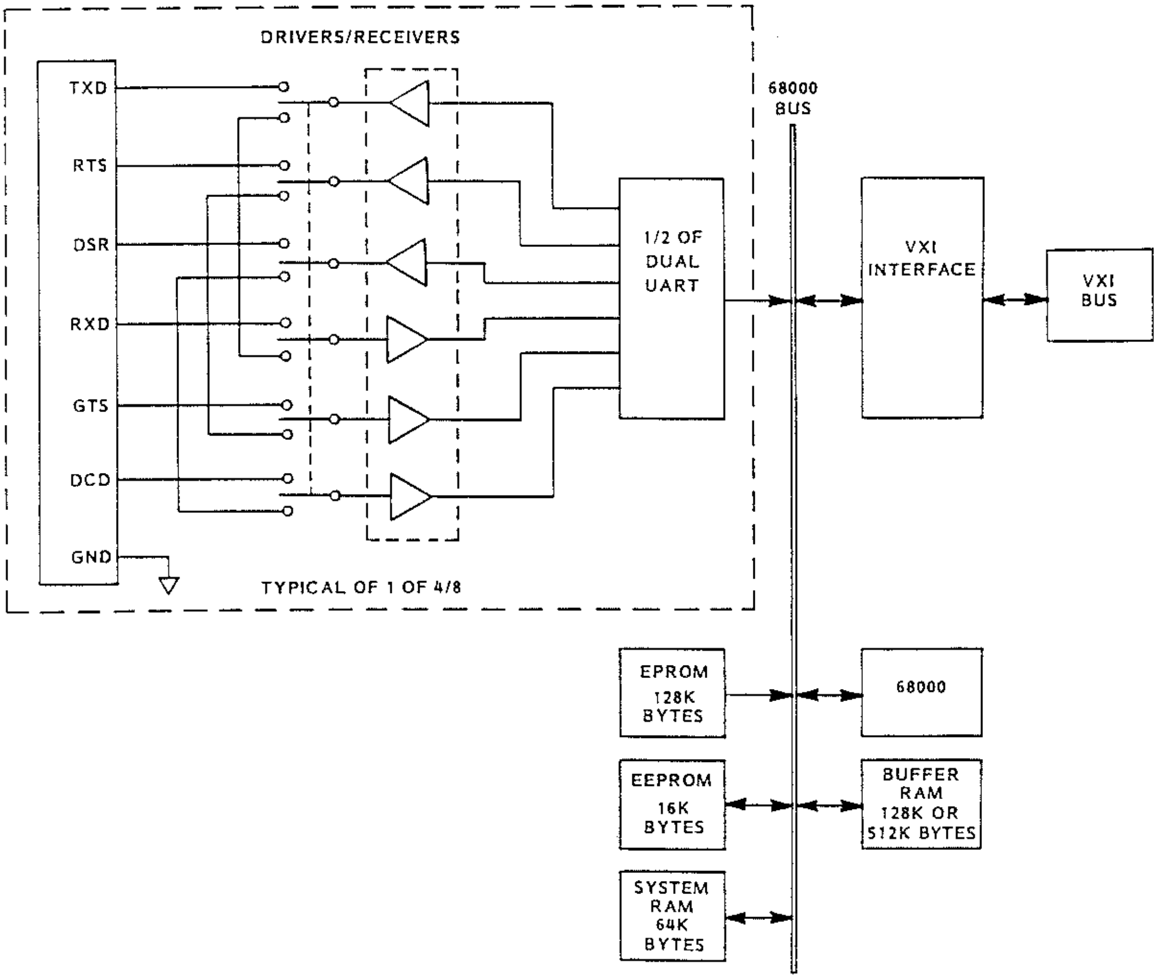


Figure 1-1 Model 6065 Block Diagram

1.3 CONFIGURATIONS

The Model 6065 is available in two versions.

Model 6065-4	4-Channel
Model 6065-8	8-Channel

1.4 SPECIFICATIONS

Buffer Size	128 k or 512 kbytes input and/or output
Serial Port	
Baud Rate (Programmable)	300, 600, 1200, 2400, 4800, 9600, 19200, 38400
Minimum Receive Jitter Tolerance	±4.5%
Data Length	5, 6, 7, 8
Parity	None, odd, even and mark or space
Stop Bits	1, 2
Handshaking	
Software	XON/XOFF
Hardware	DSR and CTS for output pacing
Power Requirements (I_{pm})	
+5 V	2 A
±12 V	100 mA
±24 V	200 mA
Weight	2.20 lb (1 kg)
Dimensions	Single Slot, C-size
VXIbus	Message-Based Revision 1.4

1.5 SAFETY

Refer to the "**FOR YOUR SAFETY**" page preceding the Table of Contents. Follow all **NOTES, CAUTIONS** and **WARNINGS** to ensure personal safety and prevent damage to the instrument.

1.6 PRODUCT SUPPORT

Racal Instruments supports the Model 6065 with Product Engineering, Service and Parts Departments. A complete listing of service centers and field representatives is provided on the last two pages of this manual.

2.1 INTRODUCTION

This section describes the unpacking, inspection, reshipment, installation and environmental requirements of the 6065 Serial Interface card.

2.2 UNPACKING AND INSPECTION

1. Prior to unpacking the module, check the exterior of the shipping carton for any signs of damage. All irregularities should be noted on the shipping bill.

2. Remove the instrument from the carton, preserving the factory packaging as much as possible.

3. Inspect the module for any defect or damage. Immediately notify the carrier if any damage is apparent.

4. Before use, have a qualified person perform a safety check.

2.3 RESHIPMENT INSTRUCTIONS

1. Use the original packaging when returning the VXI module to Racal Instruments for servicing. The shipping carton and packing material will provide the necessary support for safe reshipment. If the original packaging is unavailable, wrap the module in plastic sheeting, and use plastic spray foam to surround and protect the instrument.

2. Reship in the orginal or new shipping carton.

2.4 LOGICAL ADDRESS SETTING

The 6065 has an 8-position address switch used to set the module's logical address or set the module for dynamic configuration. The address switch is located on the "C" side of the module accessible through the sheet metal cover.

An address of 255 (FF $_{16}$ or all 1's) sets the module for dynamic configuration. In dynamic configuration, the Resource Manager sets the actual logical address of the module.

Any setting other than 255 indicates a static configuration. In a static configuration, the setting of the switches determines the address.

A switch set to the ON position produces a 1 for that bit position. A switch set to OFF produces a 0 for that bit position. Note that the most significant bit of the address is the right hand switch labeled "8" and the least significant bit of the address is the left hand switch labeled "1".

Note

Logical address 0 is not allowed.

Note

The 6065 module is shipped with the address set for dynamic configuration. Refer to your Resource Manager's manual for details about its addressing methods.

The following are example address switch settings:

1	2	3	4	5	6	7	8	
1	1	1	1	1	1	1	1	255 = Dynamic Configuration (Factory Default)

1	2	3	4	5	6	7	8	
1	0	0	0	0	0	0	0	Logical Address 1

1	2	3	4	5	6	7	8	
0	1	0	0	0	0	0	0	Logical Address 2

1	2	3	4	5	6	7	8	
0	0	1	0	0	0	0	0	Logical Address 4

1	2	3	4	5	6	7	8	
0	0	0	1	0	0	0	0	Logical Address 8

1	2	3	4	5	6	7	8	
0	0	0	0	1	0	0	0	LA 16

1	2	3	4	5	6	7	8	
0	0	0	0	0	1	0	0	LA 32

1	2	3	4	5	6	7	8	
0	0	0	0	0	0	1	0	LA 64

1	2	3	4	5	6	7	8	
0	0	0	0	0	0	0	1	LA 128

2.5 VXIBUS INTERRUPT HANDLER SETTING

One programmable interrupt line is provided on the Model 6065 module. This line is assigned by using the Assign Interrupter Line word serial protocol command (see page 182 of the *VXIbus System Specification Revision 1.4*). The Int_ID is set to 1. Normally the interrupt line is assigned by the resource manager during the power-up resource allocation.

The interrupt line is used for interrupts generated by changes in the status register. When a bit in the Service Request Enable register is a 1 and the corresponding bit in the status byte changes to a 1, the module generates a Request True event in the form of an interrupt. When all bits that cause Request True to be sent are removed, the module generates a Request False event in the form of an interrupt. It is recommended the user read Section 11 of *ANSI/IEEE-STD-488.2-1987* on device status reporting and Section E.4 of *VXIbus System Specification Revision 1.4* on Protocol Events, as well as the section on interrupts in the user's manual for the slot 0 being used.

2.6 MAINFRAME INSTALLATION

The 6065 module is ready for operation when shipped. The 6065 address switch is set to 255 at the factory to allow for dynamic configuration by the VXIbus Slot 0 Resource Manager. See Section 2.8 **LOGICAL ADDRESS SETTING** if static configuration (any switch setting other than 255) is desired.

To install the Model 6065 in a C size VXIbus chassis:

1. Ensure power on the VXI chassis is OFF.

2. Open the front cover of the chassis, if any.

3. Configure the interrupt daisy chain on the backplane so the installation slot will **not** have the slot bypassed. (Refer to the chassis operations manual for information on backplane configuration.)

4. Slide the 6065 into the desired slot with the front panel nomenclature presented correctly.

5. Seat the module in the backplane.

6. Screw in the module's retaining screws.

2.7 POWER-UP SELF-TEST INITIALIZATION

At power-up, the 6065 goes through a series of tests to assure proper operation and to establish the proper start-up state. Tests are performed on the module's ROM, RAM, non-vol, and the timer to make sure they are operating correctly.

Turn on the VXIbus chassis. The FAIL LED should come on when power is applied. At the conclusion of the self-test (less than 4.9 seconds), the FAIL LED should be extinguished. An illuminated FAIL LED after 5 seconds is an indication of a module failure. Power should be turned off. The module should be removed from the VXIbus chassis, and examined for damage or improper installation.

3.1 INTRODUCTION

The Model 6065 module is a VXIbus message-based device whose command set is compliant with the Standard Command For Programmable Instruments (SCPI) programming language. See the section on Applications for an overview of the commands and an introduction to the terminology. See the Sample Program in Appendix A for specific programming examples and command usage. Also refer to individual command descriptions.

All module commands are sent over the VXIbus backplane to the module using the VXI word serial **Byte Available** command. Each module command is terminated by setting the END bit in the **Byte Available** command with the last character of the command. All module commands may be in upper, lower or mixed case. All numbers are sent in ASCII decimal unless otherwise noted.

The module recognizes SCPI commands. SCPI is a tree-structured language based on IEEE-STD-488.2 Specifications. It utilizes the IEEE-STD-488.2 Standard command, and the device dependent commands are structured to allow multiple branches off the same trunk to be used without repeating the trunk. To use this facility, terminate one branch with a semicolon and start the next branch with a colon. As an example, receive baud rate, receive parity, receive data bits, receive stop bits and receive pacing are all branches off the **serial:receive** trunk. This makes it possible to combine several commands as follows:

```
ser2:rec:baud 1200;:par even;:bits 7;:sbit 1
```

The above command is the same as the following:

```
ser2:rec:baud 1200
ser2:rec:par even
ser2:rec:bits 7
ser2:rec:sbit 1
```

See the *Standard Commands for Programmable Instruments (SCPI)* Manual, Volume 1: Syntax & Style in Section 6 for more information.

The SCPI commands are listed in upper and lower case. Character case is used to indicate different forms of the same command. Key words can have both a short form and a long form (some commands only have one form). The short form uses just the key word characters in uppercase. The long form uses the key word characters in uppercase plus the key word characters in lowercase. Either form is acceptable. Note that there are no intermediate forms. All characters of the short form or all characters of the long form must be used. Short forms and long forms may be freely intermixed. The actual commands sent can be in upper case, lower case or mixed case (case is only used to distinguish long form and short form for the user). As an example, these commands are all correct and all have the same effect:

```
ser2:rec:baud 1200
serial2:rec:baud 1200
ser2:receive:baud 1200
serial2:receive:baud 1200
```

The following command is **not** correct because it uses part of the long form of SERial, but not all letters of the long form.

```
seri2:rec:baud 1200
```

All of the SCPI commands also have a query form unless otherwise noted. Query forms contain a question mark (?). The query form allows the system to ask what the current setting of a parameter is. The query form of the command generally replaces the parameter with the question mark. Query responses do not include the command header. This means only the parameter is returned; no part of the command is returned.

When character data is used for a parameter, both short and long forms are recognized. If the command has a query form with character response data, the short form is always returned in upper case. As an example, to find out what the current receive BAUD rate is on channel 2, use the following command:

```
ser2:rec:baud ?
```

The response could be:

```
1200
```

This tells the user that the channel 2 receive baud rate is set to 1200 baud.

Multiple commands can also be combined on one line. To do this, terminate one command with a semicolon and start the next command with a colon. As an example, channel 2 format and receive baud rate could be set as follows:

```
form:data 2 int;:ser2:rec:baud 1200
```

When combining commands, keep in mind the input buffer has a limit of 4095 characters. Command lines that are too long will generate an error and not be used.

The IEEE-STD-488.2 Common Commands can be placed anywhere set off from the rest of the command by a semicolon. They can also be placed alone on a line. For example, place the ***rst** command in front of an initialization string as follows:

```
*rst;ser2:rec:baud 1200;:par even;:bits 7;:sbit 1
```

Note that the **ser2:rec:baud 1200** command **did not** require a leading colon because there was no prior trunk of the SCPI tree.

Input Parameters

Some commands have an input parameter of **<boolean>**. The actual parameter accepted can be **on**, **off** or a number. If the parameter is a number, it is evaluated as either zero or non-zero. Zero and **off** have the same effect. Non-zero and **on** have the same effect. When the query form of the command is used, the response is always numeric ASCII. The return value is either **0** or **1**.

Some commands have optional parts and/or optional parameters. The parts between [and] are optional. If the optional part is a key word, the key word can be included or left out. If the optional part is a parameter, omitting the parameter will cause a default value to be used. The default value is 1 unless otherwise stated.

Some parameters are specified as <NRf>. <NRf> is a very general number format. It includes numbers we would call integers, numbers with decimal points, and numbers with exponential notation. Any of these forms may be used to specify the parameter. The query response of a <NRf> parameter depends on the actual command. See the example response for the specific query.

Spaces can be used freely among commands. The following commands are both correct and both have the same effect:

```
ser2     :     rec   : baud    1200

ser2:rec:baud1200
```

Some parameters are specified as <block>. <block> represents the general form of IEEE-STD-488.2 block. It specifies a group of data. Blocks can be used to pass any data including non-printable characters. The Model 6065 utilizes two different types of blocks; **definite length** arbitrary blocks and **indefinite length** arbitrary blocks.

A definite length block specifies a group of data with a known length. A definite length block has the following general form:

#xyy..yzzzzz...zzzz

The x is a single decimal digit from 1 to 9. It specifies the number of digits in the y's. The y's are a string of 1 to 9 decimal digits that represent the characters length of the data block. Note leading zeros are allowed in the y's. The z's represent the actual data.

Here are some examples of **valid** definite blocks:

#11A	A block with just the letter "A".
#12AB	A block with two letters "AB".
#19ABCDEFGHI	A block with 9 letters "ABCDEFGHI".
#210ABCDEFGHIJ	A block with 10 letters "ABCDEFGHIJ"
#9000000001A	A block with just the letter "A".

Here are some examples of **invalid** definite blocks:

#0	The first digit cannot be 0 (0 is for indefinite).
#11AB	The block says 1 character but there are two.
#AB	The character after the # must be 1 to 9.
#2AB	The y's must be decimal digits.

An indefinite length block specifies a group of data with an unknown length. An indefinite length block starts with #0 and ends with the newline (linefeed) character sent with the END indicator. If we represent data with z's and the newline with END as **NL**, an indefinite length block has the following general form:

#0zzzz...zzz**NL**

VXIbus message-based interfaces pass data with the **Byte Available** and **Byte Request** commands. Both commands pass 9 bits of information, 8 data bits and an END indication. When an indefinite length block is used to pass data, the **NL** is **not** considered part of the data.

Some examples may help to illustrate the indefinite length block. These examples are similar to the valid definite length blocks illustrated above:

#0A**NL**	A block with just the letter "A".
#0AB**NL**	A block with two letters "AB".
#0ABCDEFGHI**NL**	A block with 9 letters "ABCDEFGHI".
#0ABCDEFGHIJ**NL**	A block with 10 letters "ABCDEFGHIJ".

3.2 LOCAL COMMAND SET

This section contains a description of the individual commands recognized by the module. The commands are grouped into three categories.

The first category is IEEE-STD-488.2 Common Commands. These commands are included to comply with IEEE-STD-488.2 and SCPI. They deal with status reporting and the use of the status byte. They are also used for the control of interrupts and for device synchronization.

The second category is Required SCPI Commands. They are included to comply with SCPI. These commands deal with SCPI specific status reporting including error reporting.

The third category is Device Dependent SCPI Commands. These are used to configure and operate the instrument.

3.2.1 IEEE-STD-488.2 Common Commands

***CLS**

Clear Status Register - Clear all event registers, clear the request for OPC (Operation Complete) flag, and clear all queues (except the output queue). An example command is:

```
*cls
```

This command clears the Status register.

*ESE <mask>

Event Status Enable - Used to set the bits in the Event Status Enable (ESE) register. See ANSI/IEEE-STD-488.2-1987 Section 11.5.1 for a complete description of the ESE register. A 1 in a bit position of the ESE register enables generation of the Event Status Bit (ESB) in the Status Byte by the corresponding bit in the ESR (Event Status Register). If the ESB (Service Request Enable) is set in the SRE register, a Request True Event will be generated. See the ESR? (Event Status Register Query) command for details of the individual bits. The ESE register layout is:

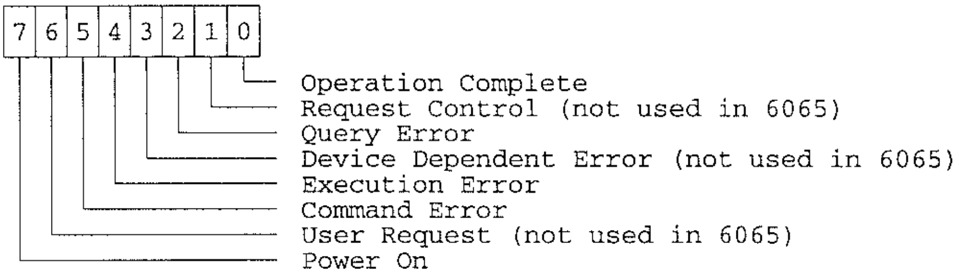

7
6
5
4
3
2
1
0
Operation Complete
Request Control (not used in 6065)
Query Error
Device Dependent Error (not used in 6065)
Execution Error
Command Error
User Request (not used in 6065)
Power On

Figure 3-1 Event Status Enable Register

An example command is:

```
*ese 36
```

This command enables the ESB in the status register to be set when there is a query error or a command error. The value 36 (00100100) comes from 4 (Query Error) + 32 (Command Error).

*ESE?

Event Status Enable Query - Queries the current contents in the Event Status Enable register. The response is a numeric ASCII value. An example command is:

```
*ese?
```

An example response is:

```
36
```

This response says the Query Error (4) and Command Error (32) events are enabled for generation of the ESB in the status register.

***ESR?**

Event Status Register Query - Queries and clears the contents in the Standard Event Status register. The response is a numeric ASCII value. This register is used in conjunction with the ESE register to generate the ESB in the Status Byte. The layout of the ESR is:

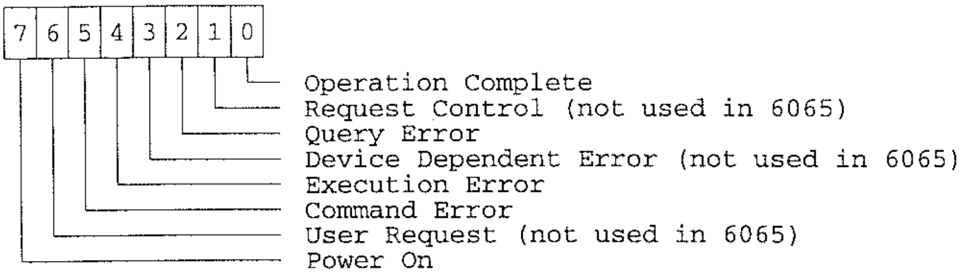

7
6
5
4
3
2
1
0
Operation Complete
Request Control (not used in 6065)
Query Error
Device Dependent Error (not used in 6065)
Execution Error
Command Error
User Request (not used in 6065)
Power On

Figure 3-2 Event Status Register

The Operation Complete bit is set by the 6065 when it receives an ***OPC** (Operation Complete) command.

The Request Control bit is not set by the 6065.

The Query Error bit is set when data is over-written in the output queue. This could occur if one query is followed by another without reading the data from the first query.

The Device Dependent Error bit is not set by the 6065.

The Execution Error bit is set when an execution error is detected. See Section 3.5 ERROR MESSAGES for a list of execution errors. Errors in the range of -200 to -299 are execution errors.

The Command Error bit is set when a command error is detected. See the section on ERROR MESSAGES for a list of command errors. Errors in the range of -100 to -199 are execution errors.

The User Request bit is not set by the 6065.

The Power-On bit is set when the module is powered-on, or it receives a reset via the VXIbus control register. Once this bit is cleared by the *ESR? command, it remains cleared.

An example command is:

```
*esr?
```

An example response is:

    4

This response means the Query Error event has occurred. A second query would produce a response of 0 because the query clears the register.

***IDN?**

Identification Query - Returns the identification string of the 6065 module. The response is character data. The response has four fields separated by commas. The first field is the name of the manufacturer. The second field is the model number. The third field is an optional serial number. If the serial number is not supplied, the third field contains a zero. The final field is the firmware revision number. An example command is:

```
*idn?
```

An example response is:

    Racal Instruments Inc.,6065-8,0,1.8

This tells the user the manufacturer is Racal Instruments. The model is 6065-8. The serial number is not used, and therefore, is zero. The firmware revision number is 1.8.

***OPC**

Operation Complete - Sets the OPC bit in the Event Status register when all pending operations have finished. Note that the 6065 never has any pending operations so it sets the OPC bit when the command is received. An example command is:

```
*opc
```

This command sets the OPC bit when all pending operations have finished.

***OPC?**

Operation Complete Query - Returns a 1 to the output queue when all pending operations have completed. Note that the 6065 never has any pending operations so it returns a 1 as soon as the command is recognized. An example command is:

```
*opc?
```

An example response is:

```
1
```

When this response is returned, it indicates all pending operations have completed.

***RCL <n>**

Recall Saved State - Recalls a previously stored Model 6065 configuration. <n> (1 to 16) is the configuration number of the desired (previously stored) set-up. During power-up, location 1 is used to set the module configuration. The following information is saved in each configuration for all channels:

FORMat:DATA
SERial:CONTrol:CTS
SERial:CONTrol:DSR
SERial:CONTrol:DTR
SERial:RECeive:BAUD
SERial:RECeive:PARity
SERial:RECeive:BITS
SERial:RECeive:SBITs
SERial:RECeive:PACE
SERial:RECeive:PACE:THReshold:STARt
SERial:RECeive:PACE:THReshold:STOP
SERial:STANdard
SERial:TRANsmit:AUTO
SERial:TRANsmit:BAUD
SERial:TRANsmit:PACE
TERMinator:LENGth or TERMinator:CHARacter
TRACe:POINts
TRIGger:AUTO
TRIGger:SEQuence:TIMer

The ***RST** command does not affect this command.

***RST**

Reset - Resets the Model 6065's hardware and firmware to a known state. The following items are set for each channel as shown:

FORMat:DATA ASCii
SERial:CONTrol:CTS 0
SERial:CONTrol:DSR 0
SERial:CONTrol:DTR OFF
SERial:RECeive:BAUD 9600
SERial:RECeive:PARity NONE
SERial:RECeive:BITS 8
SERial:RECeive:SBITs 1
SERial:RECeive:PACE NONE
SERial:RECeive:PACE:THReshold:STARt buffer length - 2K
SERial:RECeive:PACE:THReshold:STOP buffer length - 1K
SERial:STANdard setting in RCL 1
SERial:TRANsmit:AUTO 0
SERial:TRANsmit:BAUD 9600
SERial:TRANsmit:PACE NONE
TERMinator:LENGth 1
TRACe:POINts all to 1/n of total buffer size
TRIGger:AUTO 1
TRIGger:SEQuence:TIMer 0.0

***SAV <n>**

Save State - Store the present configuration for all channels in non-vol memory. <n> (1 to 16) is the configuration number used for storing the set-up. See ***RCL** for the information stored. The ***RST** command does not affect this command. It uses the **STANdard** value stored in configuration 1.

*SRE <mask>

Service Request Enable - Used to set the Service Request Enable register bits to generate a service request. The Service Request Enable register is an 8-bit register. If one of these bits is set and the corresponding bit in the Status register becomes true, a Request True event will be sent. Bit 6 (Master Summary Status) is always set by the internal software regardless of what mask value is sent. The layout of the register is:

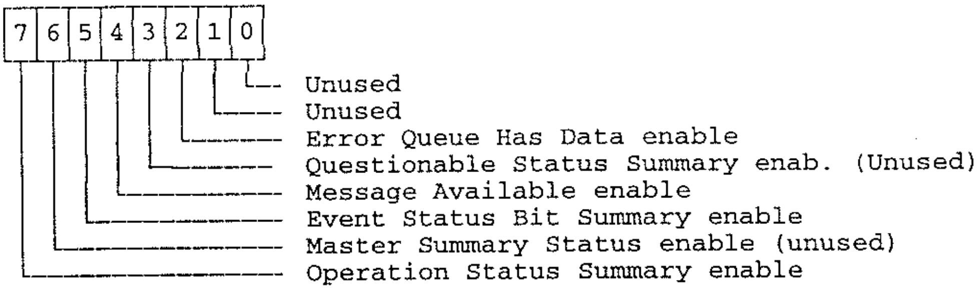

7 6 5 4 3 2 1 0
Unused
Unused
Error Queue Has Data enable
Questionable Status Summary enab. (Unused)
Message Available enable
Event Status Bit Summary enable
Master Summary Status enable (unused)
Operation Status Summary enable

Figure 3-3 Service Request Enable Register

An example command is:

```
*sre 4
```

This command enables interrupts when the Error Queue has data.

*SRE?

Service Request Enable Query - Queries the current contents in the Service Request Enable register. The response is a numeric ASCII value. See the ***SRE** command for a layout of the register. An example command is:

```
*sre?
```

An example response is:

```
68
```

This response indicates interrupts are enabled for the Error Queue (value 4) and the Master Summary Status (value 64).

***STB?**

Read Status Byte Query - Queries the current contents in the Status Byte register. The layout of the status register is:

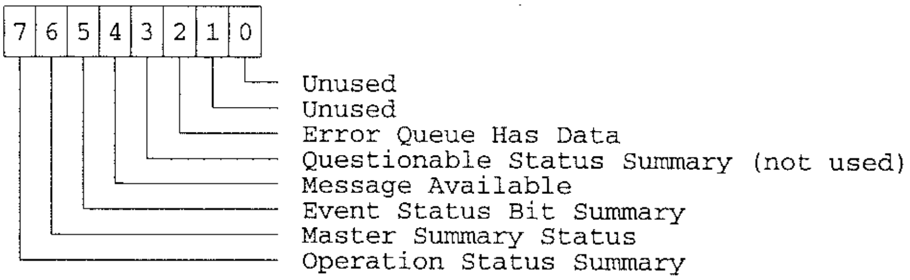


Figure 3-4 Status Byte Register

An example command is:

```
*stb?
```

An example response is:

```
16
```

This response tells the user that there is a message available in the output queue.

***TRG**

Trigger - Used to generate a trigger event. This will trigger all transmit channels set to run in block mode that are not already running with a timed trigger. See Section 3.3, BLOCK MODE for more details. An example command is:

```
*trg
```

If channel 4 was set in block mode and ready to go, this command would start channel 4 sending its transmit queue.

***TST? [<channel>]**

Self-Test Query - Performs a self-test and responds with the results. This command performs a series of tests on global resources and on individual channels. The module is isolated from outside connections by opening relays while the tests are performed. All data in the trace buffers, both transmit and receive, are lost. After the tests are completed, the module is placed in the same mode of operation as it was before the tests. The command without a channel performs the actual tests. The channel-less command must be run first to have good results. The command form with the channel reports further information about a particular channel, but does not perform any more tests.

All tests are performed regardless of the results of previous tests. The tests are:

1. Buffer RAM test
2. Determination of the number of channels installed
3. Channel tests

The channel tests are performed for each channel installed. The channels tests are:

1. RTS-CTS wrap
2. DTR-DSR wrap
3. A non-interrupt test using RS-232 mode
4. An interrupt test using RS-232 mode
5. A non-interrupt test using RS-422 mode
6. An interrupt test using RS-422 mode
7. A non-interrupt test using RS-423 mode
8. An interrupt test using RS-423 mode
9. A non-interrupt test using RS-485 mode
10. An interrupt test using RS-485 mode
11. A non-interrupt test using the UART internal wrap mode
12. An interrupt test using the UART internal wrap mode

All channel tests except the internal UART wrap tests go through the driver/receiver chips used for that mode of operation. Relays are used to wrap the drivers back to receivers. After the tests are performed, the results are analyzed for failures. The information reported from the analysis is dependent on the form of the command used.

The command form without a channel - Performs a self-test and returns a result that indicates which, if any, channels failed. A value of 0 indicates all tests passed for a particular channel. The format with no channel responds with a binary weighted decimal number. Each bit is a 1 for a channel failure and a 0 for a nonexistent or passed channel. The value has this format:

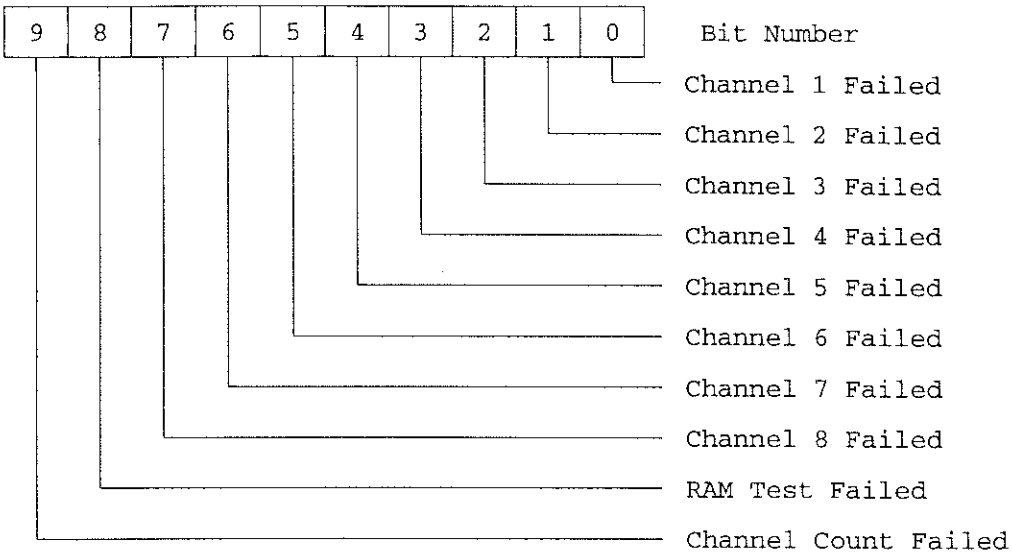


Figure 3-5 No-Channel Self-Test Results

The RAM Test Failed bit says the buffer RAM (area used for the queues) failed to pass a pseudo random pattern test, or an all zeros test. The Channel Count Failed bit says the number of installed channels detected is different from the number of channels detected at power-up.

The command form with a channel - Performs a self-test and returns a result which indicates which test failed on the selected channel. The format with a channel responds with a binary weighted decimal number. Each bit is a 1 for each test on that channel that failed:

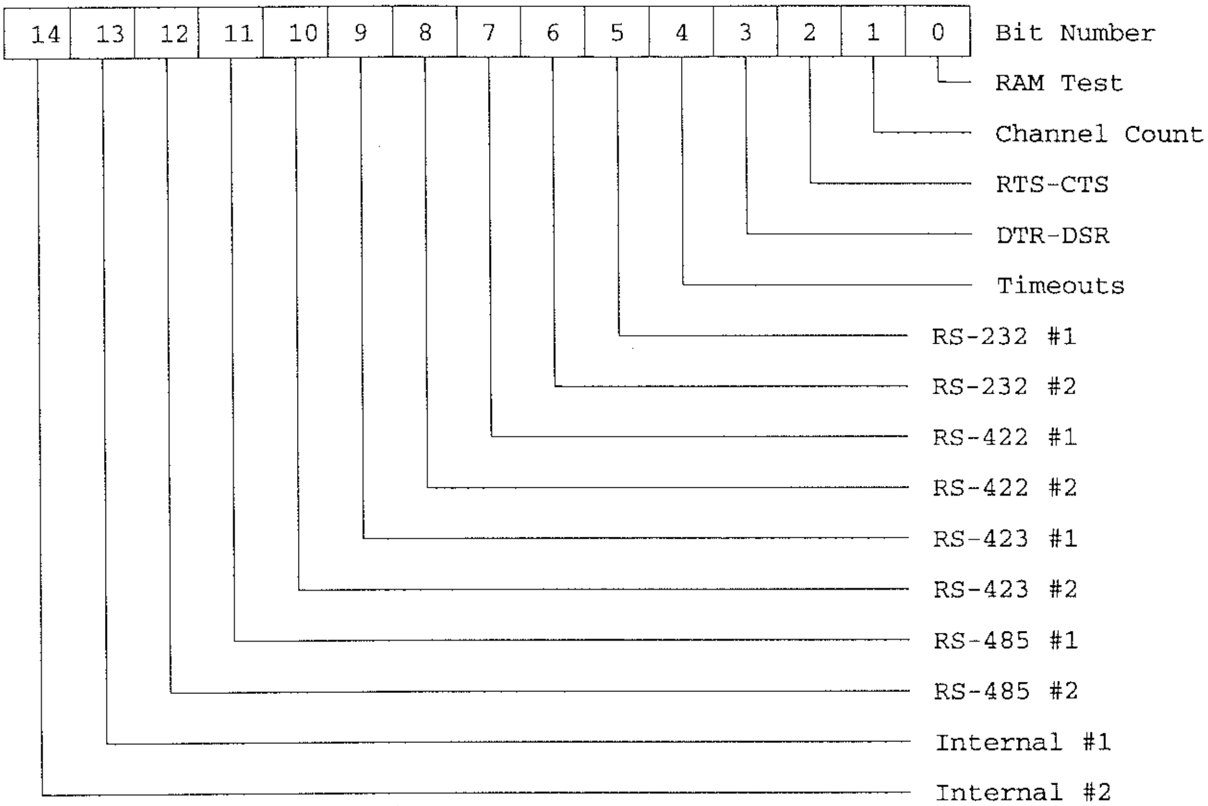


Figure 3-6 Channel Self-Test Results

The RAM Test and Channel Count bits are the same as in the channel-less form of the command. They are reported here again because they are an indication that a particular channel may not work properly.

The Timeouts bit indicates the type of failure that occurred on the channel wrap tests. It is mainly for further diagnosis of a problem.

Each electrical mode (RS-232, RS-422, RS-423, RS-485) is run with the appropriate drivers and receivers in two different tests. The first test sends and receives data on a character-by-character basis without relying on interrupts. The second test is similar, but uses interrupts for both receive and transmit.

The internal tests are the same tests performed in the different electrical modes, but use the internal loopback mode of the UART, bypassing all drivers, receivers and relays.

***WAI**

Wait To Continue - Halts execution of commands and queries until the No Operation Pending message is true. This command makes sure all previous commands have been executed before proceeding. It provides a way of synchronizing the module with its master. Note the 6065 never has any pending operations so this command is effectively a "no operation" command. An example command is:

```
*wai
```

This example commands the module to wait until all pending operations are completed.

3.2.2 Required SCPI Commands

Required SCPI commands are available in addition to the commands specifically for the 6065. Refer to the SCPI standard *Volume 1: Syntax & Style* for more information.

These commands deal with two 16-bit registers. They are the operation status register and the questionable status register. The 6065 doesn't modify any questionable status register bits. It is included for SCPI compatibility. The 6065 doesn't change any of the lower 8 bits of the operation status register. It is also included for SCPI compatibility. The layout of the lower 8 bits of the operation status register is:

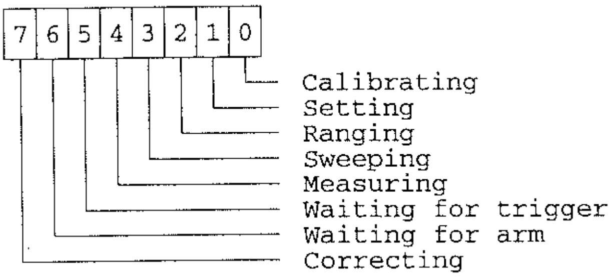


Figure 3-7 Lower 8 Bits, Operation Status Register

For more information on status data structures, the user is referred to *ANSI/IEEE-STD-488.2-1987* Section 11 Device Status Reporting.

[:]STATus:OPERation:CONDition?

Query the Operation Status Condition register. An example command is:

```
stat:oper:cond?
```

An example response is:

```
64
```

This means the module is currently waiting for an arm to occur.

[:]STATus:OPERation:ENABle <NRf>

Set the Operation Status Enable register. The limits are 0 - 32767. Note that the 6065 doesn't change any bits. This command is included for SCPI compatibility.

An example command is:

```
stat:oper:enab 0
```

This says to disable all operation status register events.

[:]STATus:OPERation:ENABle?

Query the Operation Status Enable register. The response is a numeric ASCII value. Note that the 6065 doesn't change any bits in the Operation Status register. This command is included for SCPI compatibility.

An example command is:

```
stat:oper:enab?
```

An example response is:

```
0
```

This says all operation status events are disabled.

[:]STATus:OPERation[:EVENt]?

Query the Operation Status Event register. An example command is:

```
stat:oper?
```

An example response is:

```
0
```

This means no operation events have occurred.

[:]STATus:PRESet

Preset the status registers. This command sets the following registers:

Operational Status Enable register is set to 0.

Questionable Status Enable register is set to 0.

An example command is:

```
stat:pres
```

[:]STATus:QUEStionable:CONDition?

Query the Questionable Status Condition register. An example command is:

```
stat:ques:cond?
```

An example response is:

```
0
```

[:]STATus:QUEStionable:ENABle <NRf>

Set the Questionable Status Enable register. The limits are 0 - 32767. Note that the 6065 does not modify any bits in the questionable status register. It is included for SCPI compatibility. An example command is:

```
stat:ques:enab 0
```

[:]STATus:QUEStionable:ENABle?

Query the Questionable Status Enable register. The response is a numeric ASCII value. Note that the 6065 doesn't change any bits in the Questionable Status register. This command is included for SCPI compatibility.

An example command is:

```
stat:ques:enab?
```

An example response is:

```
0
```

This says all questionable status events are disabled.

[:]STATus:QUEStionable[:EVENt]?

Query the Questionable Status Event register. An example command is:

```
stat:ques?
```

An example response is:

```
0
```

[:]SYSTem:ERRor?

Query the error queue. The error queue will maintain up to two error messages. If more than two error messages are put in the queue, an overflow condition will occur. If the queue overflows, the second and subsequent error messages are lost and an overflow message goes into the second position. See the *Standard Commands for Programmable Instruments (SCPI) Standard Volume 2: Command Reference* for details on errors and reporting. See Section 3.5 on ERROR MESSAGES for a list of errors and what they mean. An example command is:

```
syst:err?
```

This asks for the error message at the front of the error queue. An example response is:

```
0, "No error"
```

This response means no errors have occurred since the last time the error queue was cleared. Another example response is:

```
-350, "Queue overflow"
```

This response indicates more errors have occurred than the error queue could hold.

[:]SYSTem:VERSion?

Query the SCPI version number. This command returns the SCPI version number that the module complies with. The response is numeric data. An example command is:

```
syst:vers?
```

An example response is:

```
1992.0
```

3.2.3 Device Dependent SCPI Commands

[:]ABORt

Stop current block operations and all active timers; the buffers and settings are unchanged. This command is an event and has no associated, no query form and no ***RST** value. An example command is:

```
abor
```

This example says to stop current block operations.

[:]FORMat[:DATA] [<channel>] <type>

Set the data format for retrieving received characters. The actual output format is also dependent on the **TERM** commands which tells when a receive block should end. The **TERM** commands can cause blocks to be indefinite or definite. See the **TERM** commands, later in this section, for more details. This command does not allow the optional length parameter. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. Valid formats are:

ASCii Data is transferred in NR1 format with 1, 2, or 3 significant digits. Multiple numbers are separated by commas. An example output for "ABC" is:

65,66,67

INTeger Data is transferred as either a definite or an indefinite block. The length is fixed at 8 bits. An example of the indefinite block for "ABC" is:

#0ABC

An example of a definite block for "ABC" is:

#13ABC

HEXadecimal Data is encoded as a non-decimal numeric, base 16, preceded by "#H" as specified in IEEE-STD-488.2. The length is fixed at 2 digits. An example output for "ABC" is:

#H41,#H42,#H43

OCTal Data is encoded as a non-decimal numeric, base 8, preceded by "#Q" as specified in IEEE-STD-488.2. The length is fixed at 3 digits. An example output for "ABC" is:

```
#Q101,#Q102,#Q103
```

BINary Data is encoded as a non-decimal numeric, base 2, preceded by "#B" as specified in IEEE-STD-488.2. The length is fixed at 8 digits. An example output for "ABC" is:

```
#B01000001,#B01000010,#B01000011
```

PACKed PACKed is the same as INTeger above.

An example command is:

```
format 2 int
```

The example command sets the format on channel 2 to INTeger.

At ***RST**, the format for all channels is set to ASCii.

[:]FORMat[:DATA]? [<channel>]

Query the current data format for retrieving received characters. The response is character data that indicates one of the data formats. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. An example is:

```
format? 2
```

This example queries the format for channel 2. The response might be:

```
INT
```

This response indicates the format is INTeger. The ***RST** command does not directly affect query commands.

[:][SYSTem:][COMMunicate:]SERial[<channel>]:CONTrol:CTS <boolean>

Enable or disable Clear To Send (CTS) handshake mode for TRANSMISSION. The CTS signal is an input to the 6065. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The parameter **<boolean>** can take on the values described earlier (typically 0 or 1). This hardware handshake scheme will stop transmission when the CTS signal is de-asserted. When CTS is removed mid-character, transmission will continue to complete the current character. The data line then goes to its idle (marking) state waiting for CTS to be re-asserted. This mode uses no CPU processing for handshaking. It is preferred to DSR on XON handshaking only for that reason. An example command is:

```
ser2:cont:cts 1
```

This command says to use CTS handshaking to control serial port 2 transmission.

At ***RST**, this value is set to **0** - CTS handshake is disabled on all channels. While disabled, CTS is ignored and need not be connected.

[:][SYSTem:][COMMunicate:]SERial[<channel>]:CONTrol:CTS?

Query the CTS handshake mode. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The response is a numeric value of either **0** or **1**. A response value of **0** indicates the specified channel is not using CTS handshake. A return value of **1** indicates the specified channel is using CTS handshake. An example command is:

```
ser2:cont:cts?
```

This command asks if CTS handshake is used on channel 2. An example response is:

```
1
```

This response says CTS handshake is being used on channel 2. The ***RST** command does not directly affect query commands.

[:][SYSTem:][COMMunicate:]SERial[<channel>]:CONTrol:DSR <boolean>

Enable or disable Data Set Ready (DSR) handshake mode for TRANSMISSION. The DSR signal is a polled input to the 6065. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The parameter **<boolean>** can take on the values described earlier (typically **0** or **1**). This handshake scheme will stop transmission when the DSR signal is de-asserted. When DSR is removed mid-character, transmission will continue to complete the current character, and may transmit one more character as well. The data line then goes to its idle (marking) state waiting for DSR to be re-asserted. This mode uses CPU processing for polling of the handshake line. After DSR is re-asserted, there is a time delay before transmission resumes. DSR and XON handshaking take CPU processing power to complete. Because of this, CTS is the preferred handshaking mode. An example command is:

```
ser3:cont:dsr 1
```

This command says to use CTS handshaking to control serial port 2 transmission.

At ***RST**, this value is set to **0** - DSR handshake is disabled on all channels. While disabled, DSR is ignored and need not be connected.

[:][SYSTem:][COMMunicate:]SERial[<channel>]:CONTrol:DSR?

Query the DSR handshake mode. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The response is a numeric value of either **0** or **1**. A return value of **0** indicates the specified channel is not using DSR handshake. A return value of **1** indicates the specified channel is using DSR handshake. An example command is:

```
ser3:cont:dsr?
```

This command asks if DSR handshake is used on channel 3. An example response is:

```
1
```

This response says DSR handshake is being used on channel 3. The ***RST** command does not directly affect query commands.

[:][SYSTem:][COMMunicate:]SERial[<channel>]:CONTrol:DTR ON|OFF|STANdard|IBFull

Set the Data Terminal Ready (DTR) handshake mode for RECEPTION. The DTR signal is an output of the 6065. This mode is associated with the receive buffer for the channel. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. Valid DTR modes are:

ON	The ON parameter indicates the DTR output is asserted.
OFF	The OFF parameter indicates the DTR output is un-asserted.
STANdard	The STANdard mode indicates the DTR output is to follow the RS-232 standard. DTR is asserted when the receiver is ready to accept data and un-asserted when the receiver is not ready to accept data. When the receiver is offline, the DTR signal is un-asserted.
IBFull	The IBFull parameter indicates the DTR

An example command is:

```
ser3:cont:dtr 1
```

This command says to use CTS handshaking to control serial port 2 transmission.

At ***RST**, this value is set to **0** - DTR handshake is disabled on all channels. While disabled, DTR is ignored and need not be connected.

[:][SYSTem:][COMMunicate:]SERial[<channel>]:CONTrol:DTR?

Query the DTR handshake mode. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The response is character data that indicates the current DTR handshake mode of the selected channel. Valid responses are: ON, OFF, STAN, and IBF. An example command is:

```
ser1:cont:dtr?
```

This command asks what the current DTR mode is of channel 1. An example response is:

```
OFF
```

This response says the current DTR mode of channel 1 is off. The ***RST** command does not directly affect query commands.

[:][SYSTem:][COMMunicate:]SERial[<channel>]:CONTrol:RTS <ON|OFF|STANdard|IBFull|RFR>

Set the Request To Send (RTS) handshake mode. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. Valid RTS modes are:

ON — The ON parameter indicates that RTS output is asserted.

OFF — The OFF parameter indicates that RTS output is un-asserted.

STANard — The STANDARD mode indicates RTS is to follow the RS-232 standard. It is used to indicate transmit mode activity. RTS is asserted before any transmission. It continues to be asserted while the characters are sent. RTS is de-asserted after the last character is sent out of the transmitter. RTS could become de-asserted as soon as one bit time after the last stop bit.

IBFull — This parameter sets the RTS output for use in the receive hardware handshaking mode. RTS is asserted when the receive buffer is ready to receive. The number of characters in the buffer is compared with the STOP and START thresholds. These thresholds are set independently. As the number of characters in the receive buffer increases to the STOP threshold, RTS is de-asserted. RTS remains de-asserted until the number of characters in the buffer decrease to the START threshold. When the START threshold is reached RTS is re-asserted, and the receive buffer is again ready to receive. See the command **ser:rec:pace:threshold** (START and STOP) for details.

RFR — RFR stands for Ready For Receive. It is a synonym for IBFull.

An example command is:

```
ser4:cont:rts ibf
```

This command sets channel 4 to use the IBFull mode of handshaking for the RTS line. The ***RST** command sets the RTS handshake mode to OFF for all channels.

[:][SYSTem:][COMMunicate:]SERial[<channel>]:CONTrol:RTS?

Query the RTS handshake mode on a specified channel. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The response is character data that indicates the RTS handshake mode. An example of the command is:

```
ser:cont:rts?
```

This command asks what RTS handshake mode is used on the default channel 1. An example response is:

OFF

This says RTS handshake is not being used on channel 1. The ***RST** command does not directly affect query commands.

[:][SYSTem:][COMMunicate:]SERial[<channel>][:RECeive]:BAUD <baud_rate>

Set the receive baud rate on a specified channel. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. Valid values for **<baud_rate>** are 300, 600, 1200, 2400, 4800, 9600, 19200 and 38400. An example command is:

```
ser4:baud 38400
```

This example sets the receive baud rate on channel 4 to 38400. The ***RST** command causes the receive baud rate on all channels to be set to 9600 baud.

[:][SYSTem:][COMMunicate:]SERial[<channel>][:RECeive]:BAUD?

Query the receive baud rate on a specified channel. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The response is a numeric value that is one of the valid baud rates. An example command is:

```
ser3:baud?
```

This example asks what the receive baud rate is on channel 3. An example response is:

38400

This response says the baud receive baud rate on channel 3 is 38400. The ***RST** command does not directly affect query commands.

[:][SYSTem:][COMMunicate:]SERial[<channel>][:RECeive]:BITS <bits>

Set the number of data bits. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. Although this is under receive, it actually sets both transmit and receive number of bits. Valid values for **<bits>** are 5, 6, 7, and 8. An example command is:

```
ser1:bits 8
```

This example sets the number of transmit and receive bits to 8 for channel 1. The ***RST** command causes bits on all channels to be set to 8.

[:][SYSTem:][COMMunicate:]SERial[<channel>][:RECeive]:BITS?

Query the number of data bits. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The response is a numeric value that is one of the valid number of data bits. An example command is:

```
ser1:bits?
```

This example asks what the current number of data bits is for channel 1. An example response is:

```
8
```

The response says the number of data bits on channel 1 is 8 bits. The ***RST** command does not directly affect query commands.

[:][SYSTem:][COMMunicate:]SERial[<channel>][:RECeive]:PACE?

Query the receive software handshake mode. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The response is character data that indicates the software handshake mode for the indicated channel. Valid responses are XON and NONE. An example command is:

```
ser3:pace?
```

This command asks what the current software handshake mode is for channel 3. An example response is:

```
NONE
```

This response says there is no software handshaking being used on channel 3. The ***RST** command does not directly affect query commands.

[:][SYSTem:][COMMunicate:]SERial[<channel>][:RECeive]:PACE:THReshold:STARt <NRf>

Set the number of characters in the receive buffer to enable data handshaking. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The valid range for **<NRf>** is 1 to (buffer size - 3). Buffer size is programmed using [:]TRACe:POINts. This sets the point where handshaking will begin. Normally, the **STARt** threshold for a channel is set higher than the **STOP** threshold. If **ser:pace xon** has been selected for this channel, it is the point where XOFF is sent. If **ser:cont:dtr** has been set to **IBFull**, this is the point where DTR is de-asserted. If **ser:cont:rts** has been set to **IBFull**, this is the point where RTS is de-asserted. If an attempt is made to set the start point outside the valid range, an error message is issued and the start point is set to 1/3 of the buffer size. An example command is:

```
ser3:pace:thr:star 999
```

This example sets the start threshold on channel 3 to 999. If the channel 3 receiver ever reaches the point where 999 characters are in the queue, any requested actions (i.e. XOFF, DTR or RTS) will take place. At ***RST**, all channels have their **STARt** threshold set to buffer size -1024. Note that the buffer size is dependent on the number of channels and the size of buffer RAM. It can be set with the **trac:poin** command. See the **trac:poin** command for buffer size details.

[:][SYSTem:][COMMunicate:]SERial[<channel>][:RECeive]:PACE:THReshold:STARt?

Query the pace handshaking threshold start point. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The response is a numeric ASCII value. An example command is:

```
ser3:pace:thr:star?
```

This asks what the start threshold is on channel 3. An example response is:

```
999
```

This says the start threshold on channel 3 is set to 999. The ***RST** command does not directly affect query commands.

[:][SYSTem:][COMMunicate:]SERial[<channel>][:RECeive]:PACE:THReshold:STOP <NRf>

Set the number of characters in the receive buffer to disable data handshaking. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The valid range for **<NRf>** is 1 to buffer size -1. This sets the point where handshaking will end. Normally, the **STOP** threshold is set lower than the **STARt** threshold for a channel. If **ser:pace xon** has been selected for this channel, it is the point where XON is sent. If **ser:cont:dtr** has been set to **IBFull**, this is the point where DTR is re-asserted. If **ser:cont:rts** has been set to **IBFull**, this is the point where RTS is re-asserted. If an attempt is made to set the start point outside the valid range, an error message is issued and the start point is set to 2/3 of the buffer size. An example command is:

```
ser3:pace:thr:stop 500
```

This example sets the stop threshold on channel 3 to 500. If the channel 3 receive queue ever drops to the point where 500 characters are in the queue, any requested actions (i.e. XOFF, DTR or RTS) will take place. At ***RST**, all channels have their **STOP** threshold set to buffer size -2048. Note that the buffer size is dependent on the number of channels and the size of buffer RAM. It can be set with the **trac:poin** command. See the **trac:poin** command for buffer size details.

[:][SYSTem:][COMMunicate:]SERial[<channel>][:RECeive]:PACE:THReshold:STOP?

Query the pace handshaking threshold stop point. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The response is a numeric ASCII value. An example command is:

```
ser3:pace:thr:stop?
```

This asks what the stop threshold is on channel 3. An example response is:

```
500
```

This says the stop threshold on channel 3 is set to 500. The ***RST** command does not directly affect query commands.

[:][SYSTem:][COMMunicate:]SERial[<channel>][:RECeive]:PACE <XON|NONE>

Set the receive software handshake mode. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. If the software handshake mode is set to NONE, no software handshaking is done. If the software handshake mode is set to XON, the stop and start thresholds are used to determine when to send the XON character (17 decimal) and when to send the XOFF character (19 decimal). As characters are received, the receive buffer fills. XOFF is sent when the receive buffer reaches the stop threshold. As characters are retrieved from the receive buffer, the number of characters in the receive buffer drop. When the number of characters drops below the start threshold, the XON character is sent. An example command is:

```
ser3:pace xon
```

This example sets the pace mode to XON for channel 3. XON and XOFF characters will be sent to try and regulate the quantity of characters in receive buffer 3 (RCH3). At ***RST**, all channels have their software pace mode set to NONE. Also see the commands for setting the thresholds.

[:][SYSTem:][COMMunicate:]SERial[<channel>][:RECeive]:PARity[:TYPE] <EVEN|ODD|NONE|IGNore|ZERO|ONE>

Set the parity mode. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. Although this is under receive, it actually sets both transmit and receive parity. Valid types are:

EVEN	Even parity is sent with every character. Receive characters are checked for even parity.
ODD	Odd parity is sent with every character. Receive characters are checked for odd parity.
NONE	No parity is sent with the characters. No parity is checked on receive characters. Note that if the characters being received do have parity, it is probable this will show up as framing errors.
IGNore	This retains the previous parity mode for sending characters. Parity errors are ignored on receive (the user never sees the errors).
ZERO	A parity bit of 0 is sent with every character. Receive characters are checked for a 0 parity bit.
ONE	A parity bit of 1 is sent with every character. Receive characters are checked for a 1 parity bit.

An example command is:

```
ser4:par odd
```

This example sets the receive and transmit parity of channel 4 to odd parity. The ***RST** command causes parity on all channels to be set to NONE.

[:][SYSTem:][COMMunicate:]SERial[<channel>][:RECeive]:PARity[:TYPE]?

Query the parity mode. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The response is character data that indicates the current parity mode of the selected channel. Valid responses are: EVEN, ODD, NONE, IGN, ZERO, and ONE. An example command is:

```
ser4:par?
```

This command asks what the current parity mode is of channel 4. An example response is:

```
ODD
```

This response says the current parity mode of channel 4 is odd. The ***RST** command does not directly affect query commands.

[:][SYSTem:][COMMunicate:]SERial[<channel>][:RECeive]:SBITs <bits>

Set the number of stop bits. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. Although this is under receive, it actually sets both transmit and receive number of stop bits. Valid values for **<bits>** are 1 and 2. An example command is:

```
ser2:sbit 2
```

This example sets the number of stop bits on both transmit and receive. The ***RST** command causes the number of stop bits to be set to 1 on all channels.

[:][SYSTem:][COMMunicate:]SERial[<channel>][:RECeive]:SBITs?

Query the number of stop bits. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The response is a numeric value that is one of the valid number of stop bits. An example command is:

```
ser2:sbit?
```

This example asks what the current number of stop bits is for channel 2. An example response is:

```
1
```

The response says the number of stop bits on channel 2 is 1 bit. The ***RST** command does not directly affect query commands.

[:][SYSTem:][COMMunicate:]SERial[<channel>]:STANdard <standard>

Set the electrical interface standard to use. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The valid values for **<standard>** are 232, 422, 423, 485. If **<standard>** is set to 232 this command configures the indicated channel for the RS-232 specification. If **<standard>** is set to 422, this command configures the indicated channel for the RS-422 specification. If **<standard>** is set to 423, this command configures the indicated channel for the RS-423 specification. If **<standard>** is set to 485, this command configures the indicated channel for the RS-485 specification. An example command is:

```
ser4:stan 422
```

This example sets channel 4 to the RS-422 standard interface. At ***RST**, all channels default to the standards saved in non-vol location 1 (the factory set values in non-vol location 1 are all RS-232).

[:][SYSTem:][COMMunicate:]SERial[<channel>]:STANdard?

Query the current configuration standard for the indicated channel. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The response is a numeric ASCII value. Valid responses are 232, 422, 423, or 485. An example command is:

```
ser4:stan?
```

This asks what the current configuration standard is on channel 4. An example response is:

```
422
```

This says channel 4 is set for the RS-422 standard. The ***RST** command does not directly affect query commands.

[:][SYSTem:][COMMunicate:]SERial[<channel>]:TRANSmit:AUTO <boolean>

Couple or uncouple the transmit baud rate to the receive baud rate. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. When **<boolean>** is **ON** or **1**, the transmit baud rate is set to the same rate as the receive baud rate. If the receive rate is changed, so is the transmit rate. If **<boolean>** is **OFF** or **0**, the transmit baud rate doesn't change when the receive rate is changed. Note that setting the transmit baud rate automatically turns coupling off for the specified channel. Note that the number of data bits and the channel parity are always the same for both transmit and receive, and are set with **[RECeive]** commands. An example command is:

```
ser2:tran:auto 0
```

This example uncouples the channel 2 transmit baud rate from the channel 2 receive baud rate. At ***RST**, all channels default to coupled (auto 1).

[:][SYSTem:][COMMunicate:]SERial[<channel>]:TRANSmit:AUTO?

Query coupling of the transmit and receive baud rates for the indicated channel. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The response is a numeric ASCII value. Valid responses are 0 and 1. An example command is:

```
ser2:tran:auto?
```

This asks what the current coupling is between transmit and receive baud rates on channel 2. An example response is:

```
0
```

This says channel 2 doesn't have receive and transmit baud rates coupled. The ***RST** command does not directly affect query commands.

[:][SYSTem:][COMMunicate:]SERial[<channel>]:TRANSmit:BAUD <baud_rate>

Set the transmit baud rate on a specified channel. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. Valid values for <baud_rate> are 300, 600, 1200, 2400, 4800, 9600, 19200 and 38400. Note that setting a transmit baud rate automatically turns off the auto coupling between transmit and receive baud rates. An example command is:

```
ser3:tran:baud 19200
```

This example sets the transmit baud rate on channel 3 to 19200 baud. At ***RST**, all channels default to 9600 with auto coupling to the receive baud rates.

[:][SYSTem:][COMMunicate:]SERial[<channel>]:TRANsmit:BAUD?

Query the transmit baud rate on the indicated channel. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The response is a numeric ASCII value. Valid responses are 300, 600, 1200, 2400, 4800, 9600, 19200, 38400. An example command is:

```
ser3:tran:baud?
```

This asks what the current transmit baud rate is on channel 3. An example response is:

```
19200
```

This says channel 3 transmit baud rate is set to 19200. The ***RST** command does not directly affect query commands.

[:][SYSTem:][COMMunicate:]SERial[<channel>]:TRANsmit:PACE <XON|NONE>

Set the transmit software handshake mode. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. Transmit pacing uses XON and XOFF characters received on the receive side of a channel to pace data being transmitted on the transmit side. When pace is set to XON, the receive side monitors the received characters looking for XON and XOFF. When an XOFF is received, the transmitter stops transmitting. After the transmitter has been stopped by XOFF, receiving an XON character will start the transmitter transmitting again. An example command is:

```
ser4:tran:pace xon
```

This example sets the transmit software handshake mode to use XON/XOFF on the receive channel to page the transmitter. At ***RST**, all channels default to **NONE**.

[:][SYSTem:][COMMunicate:]SERial[<channel>]:TRANsmit:PACE?

Query the transmit software handshaking mode. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The responses are XON and NONE. An example command is:

```
ser4:tran:pace?
```

This asks what the current transmit software handshake mode is on channel 4. An example response is:

```
XON
```

This says channel 4 transmit software handshaking is active. The ***RST** command does not directly affect query commands.

[:]TERMinator:CHARacter [<channel>] <character_number>

Set the character to use for end of line recognition. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The valid range of <character_number> is 0 to 255. This tells the 6065 what character to use to recognize the end of a record. Note that setting a termination character automatically sets the termination length to **0**. An example command is:

```
term:char 2 10
```

This example sets the termination character on channel 2 to 10, an ASCII line-feed character. At ***RST**, all channels default to a termination length of **1** which turns termination character off.

[:]TERMinator:CHARacter? [<channel>]

Query the character to use for end of line recognition. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The responses are: OFF or an ASCII numeric value in the range of 0 to 255. An example command is:

```
term:char? 2
```

This asks what the current termination character is for channel 2. An example response is:

OFF

This says channel 2 is not using any termination character at this time. Another example response is:

10

This says channel 2 is using an ASCII line-feed character (10) as the termination character, The ***RST** command does not directly affect query commands.

[:]TERMinator:LENGth [<channel>] <length>

Set the character count to use for end of line recognition. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The valid range of <length> is 0 to length of buffer -3. The length tells the 6065 how many characters to accumulate in a receive queue before returning any data. Note that setting a termination length of 0 allows all characters in the receive queue to be returned. Note that setting a termination length automatically turns off the termination character. An example command is:

```
term:leng 3 15
```

This example sets the termination length on channel 3 to 15 characters. At ***RST**, all channels default to a termination length of **1** which turns termination character off.

[:]TERMinator:LENGth? [<channel>]

Query the character count to use for end of line recognition. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The response is a numeric ASCII value. Valid values are 0 to buffer length -3. An example command is:

```
term:LENG? 3
```

This asks what the current termination length is for channel 3. An example response is:

```
0
```

This says channel 3 will return all character in the receive queue. Another example response is:

```
15
```

This says channel 3 will return characters in groups of 15 characters. The ***RST** command does not directly affect query commands.

[:]TRACe:DATA <trace_name>,(block>|<NRf>{,<NRf>})

Load data into the specified transmit queue. <trace_name> selects which channel's transmit queue gets loaded. Valid values for <trace_name> on a 4-channel card are TCH1, TCH2, TCH3 and TCH4. Valid values for <trace_name> on an 8-channel card are the same as a 4-channel card plus TCH5, TCH6, TCH7 and TCH8. Data can be loaded either as a block or as a series of numbers. See the description of BLOCK MODE in Section 3.3 for more details on blocks. If data is loaded as a series of numbers, the numbers are sequentially placed in the queue. An example command using a block is:

```
trac:data tch1,#13ABC
```

This command loads the characters ABC into the transmit queue of channel 1. It uses a block to load the characters. Another example command is:

```
trac:data tch1,65,66,67
```

This example loads the characters ABC into the transmit queue of channel 1. It uses a series of numbers to load the characters. The ***RST** command does not affect the data in the queues.

[:]TRACe:DATA? <trace_name>

Retrieve the characters in the specified input queue. <trace_name> selects which channel's receive queue the characters are retrieved from. Valid values for <trace_name> on a 4-channel card are RCH1, RCH2, RCH3 and RCH4. Valid values for <trace_name> on an 8-channel card are the same as a 4-channel card plus RCH5, RCH6, RCH7 and RCH8. The format of the data returned can vary. See the **form:data** command for more details. The amount of data returned depends on the **term:char** and **term:leng** commands. See those commands and Section 3.4 TERMINATION CHARACTERS AND TERMINATION LENGTH for more details on how the end of data is determined. An example command is:

```
trac:data? rch1
```

This asks for characters from the receive queue of channel 1. An example response is:

```
65,66,67
```

This says 3 characters were returned, ABC. The data format was ASC. Another example response is:

```
#13ABC
```

This says 3 characters were returned and the data format was INT with a fixed data length of 3. The ***RST** command does not directly affect query commands.

[:]TRACe:DATA:LENGth? <trace_name>

Query the number of characters in one of the channel queues. <trace_name> selects which channel's queue gets checked for length. Valid values for <trace_name> on a 4-channel card are RCH1, RCH2, RCH3, RCH4, TCH1, TCH2, TCH3 and TCH4. Valid values for <trace_name> on an 8-channel card are the same as a 4-channel card plus RCH5, RCH6, RCH7, RCH8, TCH5, TCH6, TCH7 and TCH8. The response is a numeric ASCII value. Valid values are 0 to buffer length. An example command is:

```
trac:data:leng? rch2
```

This asks how many characters are in channel 2's receive buffer. An example response is:

```
0
```

This says there are no characters in channel 2's receive buffer. Another example response is:

```
123
```

This says there are 123 characters in channel 2's receive buffer. The ***RST** command does not directly affect query commands.

[:]TRACe:FREE? <trace_name>

Query the amount of buffer memory which is unused in a queue. <trace_name> selects which channel's queue gets checked. Valid values for <trace_name> on a 4-channel card are RCH1, RCH2, RCH3, RCH4, TCH1, TCH2, TCH3 and TCH4. Valid values for <trace_name> on an 8-channel card are the same as a 4-channel card plus RCH5, RCH6, RCH7, RCH8, TCH5, TCH6, TCH7 and TCH8. The response is a numeric ASCII value. Valid values are 0 to the size of the queue. The returned value is the number of bytes not filled. Note that each character takes up two bytes in a queue. An example command is:

```
trac:free? tch1
```

This asks what amount of memory is not used by the transmit queue for channel 1. An example response is:

```
0
```

This says the transmit queue for channel 1 is full, no more characters can be added to the queue. Another example response is:

```
8192
```

This says there is enough buffer memory to add 4096 (8192 / 2 = 4096) more characters to the transmit queue for channel 1. The ***RST** command does not directly affect query commands.

[:]TRACe:POINts <trace_name>,<points>

Set the size of a transmit or receive queue. Valid values for <trace_name> on a 4-channel card are RCH1, RCH2, RCH3, RCH4, TCH1, TCH2, TCH3 and TCH4. Valid values for <trace_name> on an 8-channel card are the same as a 4-channel card plus RCH5, RCH6, RCH7, RCH8, TCH5, TCH6, TCH7 and TCH8. The valid range of <points> is 2 to the size of buffer space. Note that all buffers are re-allocated when a **trac:poin** command is received. This should not be used when the module is actively transmitting or receiving. Note that the **<points>** is specified in bytes and each character in a queue requires two bytes. If more memory is requested than is available, an error is reported and latter buffers are shorted. It is recommended that small buffers be allocated first. This will avoid unnecessary error messages by freeing up space before allocating larger buffers. An example command is:

```
trac:poin rch1,500
```

This example sets the size of the receive queue for channel 1 to 500 bytes which is room for 250 characters. At ***RST**, all buffer memory is allocated equally to all queues. The actual size is dependent on the number of channels on the board and the amount of buffer space available:

Models	Total Buffer Space	Bytes	Characters
6065-4	128k	16k	8k
6065-8	128k	8k	4k
6065-4	512k	64k	32k
6065-8	512k	32k	16k

[:]TRACe:POINts? <trace_name>

Query the size of a transmit or receive queue. Valid values for <trace_name> on a 4-channel card are RCH1, RCH2, RCH3, RCH4, TCH1, TCH2, TCH3 and TCH4. Valid values for <trace_name> on an 8-channel card are the same as a 4-channel card plus RCH5, RCH6, RCH7, RCH8, TCH5, TCH6, TCH7 and TCH8. The response is a numeric ASCII value that indicates the number of bytes allocated to a receive or transmit queue. Note that two bytes are required for every character in a queue. An example command is:

```
trac:poin? rch1
```

This asks how many bytes are allocated to the receive queue for channel 1. An example response is:

```
500
```

This says the channel 1 receive queue has 500 bytes allocated to it. That is enough to handle 250 characters. The ***RST** command does not directly affect query commands.

[:]TRIGger:AUTO [<channel>] <boolean>

Set the mode of operation of a channel to either character mode or block mode. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The valid range for **<channel>** is 1 to 4 on a 4-channel board and 1 to 8 on an 8-channel board. When set to 1 or ON, the channel operates in the character mode. Each character is sent as soon as possible after being placed in the transmit queue, i.e. it is automatically sent. When set to 0 or OFF, the channel operates in the block mode. Characters are not sent but accumulate in the transmit queue. When the channel is triggered, the characters in the transmit queue are sent. The characters are still retained in the transmit queue and can be sent over and over. The source of the trigger can be either the timer or from a command. Either the ***trg** or the **trig** command can trigger the transmission. See the section on BLOCK MODE for more details. An example command is:

```
trig:auto 2 0
```

This example sets channel 2 to block mode. The ***RST** command sets all channels to 1 - character mode.

[:]TRIGger:AUTO? [<channel>]

Query the character/block mode of operation of a channel. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The valid range for **<channel>** is 1 to 4 on a 4-channel board and 1 to 8 on an 8-channel board. The response is a numeric ASCII value. Valid responses are 0 and 1. A response of 1 indicates the channel is in character mode. A response of 0 indicates the channel is in block mode. An example command is:

```
trig:auto? 2
```

This asks what the mode is of channel 2. An example response is:

```
0
```

This response says channel 2 is operating in block mode. The ***RST** command does not directly affect query commands.

[:]TRIGger[:IMMediate]

Trigger all channels for block transmission. This command triggers all transmit channels set to run in block mode that are not already running with a timed trigger. This is exactly the same as the ***trg** command. See the **trig:auto** command and the section on BLOCK MODE for more details. An example command is:

```
trig
```

This example starts all channels set to block mode that are not already running. The ***RST** command stops all block mode transmissions but does not directly affect this command.

[:]TRIGger[:IMMediate] <channel>

Trigger one channel for block transmission. The valid range for **<channel>** is 1 to 4 on a 4-channel board and 1 to 8 on an 8-channel board. This command triggers one transmit channel set to run in block mode. If the channel is already running an error is generated. See the **trig:auto** command and the section on BLOCK MODE for more details. An example command is:

```
trig 2
```

This example starts a channel 2 block mode transmission. The ***RST** command stops all block mode transmissions but does not directly affect this command.

[:]TRIGger:SEQuence:SOURce [<channel>] IMMediate|TIMer

Set the trigger source for a block mode transmit channel. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The valid range for **<channel>** is 1 to 4 on a 4-channel board and 1 to 8 on an 8-channel board. This only has an effect on block mode channels. Note that block mode is set by the **trig:auto <channel> 0** command. When **imm** is set, the transmit operation is started by a **trig** or ***trg** command. When **tim** is set, the timer value controls the trigger. See the BLOCK MODE section for more details. An example command is:

```
trig:seq:sour 2 imm
```

This example sets channel 2 so transmission will be started by a **trig 2** command. Here is another example:

```
trig:seq:sour 3 tim
```

This example sets channel 3 so transmission will be started by the timer. The ***RST** command sets the source for all triggers to immediate.

[:]TRIGger:SEQuence:SOURce? [<channel>]

Query the trigger source for a block mode channel. If the optional **<channel>** parameter is not supplied the channel defaults to channel 1. The valid range for **<channel>** is 1 to 4 on a 4-channel board and 1 to 8 on an 8-channel board. The returned value is either **IMM** or **TIM**. The return value of **IMM** means the specified channel will be triggered by commands and not by the timer. The return value of **TIM** means the timer will trigger the specified channel. An example command is:

```
trig:seq:sour? 2
```

This asks what is the trigger source for channel 2. An example response is:

```
IMM
```

This response says channel 2 will be trigger by commands and not by the timer. The ***RST** command does not directly affect query commands.

[:]TRIGger:SEQuence:TIMer [<channel>] <time_value>

Set the time interval to be used for resending blocks of data. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The valid range for **<channel>** is 1 to 4 on a 4-channel board and 1 to 8 on an 8-channel board. The **<time_value>** parameter sets the time interval between the starts of block transmissions. The valid range of **<time_value>** is 0.001 to 2147.483 seconds. The value of 0 is also allowed. Setting the value to zero turns the timer off. Turning the timer off is a way of gracefully stopping transmissions because the current transmission will continue but no new transmission will be started. This only has an effect on block mode channels. Note that block mode is set by the **trig:auto <channel> 0** command. When **imm** is set, the transmit operation is started by a **trig** or ***trg** command. When **tim** is set, the timer value controls the trigger. See the BLOCK MODE section for more details. An example command is:

```
trig:seq:tim 3 .1
```

This example sets the channel 3 timer so transmission will be repeated every 100 mSec. Here is another example:

```
trig:seq:tim 3 0
```

This example could be used to stop channel 3 transmissions after the current transmission is finished. The ***RST** command sets all timer values to zero.

[:]TRIGger:SEQuence:TIMer? [<channel>]

Query the time interval to be used for resending blocks of data. If the optional **<channel>** parameter is not supplied, the channel defaults to channel 1. The valid range for **<channel>** is 1 to 4 on a 4-channel board and 1 to 8 on an 8-channel board. The response is a numeric ASCII value that represents the number of seconds between the start of block transmissions. An example command is:

```
trig:seq:tim? 3
```

This asks what is the time interval between the start of block transmissions for channel 3. An example response is:

```
0.100000
```

This response says channel 3 will resend blocks every 100 mSec, The ***RST** command does not directly affect query commands.

3.3 BLOCK MODE

Transmit channels can be in one of two major modes; character mode or block mode. In character mode, characters start transmitting as soon as they are placed in the transmit queue. Once a character is transmitted, it is no longer available. This mode of operation would be used when interacting with other instruments and where the data changes constantly. Block mode does not send the characters in the transmit queue until it is commanded to do so. The characters in the queue can be transmitted over and over on command. This might be used to stimulate a device over and over again. It might also be used to simulate a device that repeatedly sent the same data over and over again.

To place a channel in the block mode, use the **trig:auto** command to set auto mode **OFF** or **0**. Using this command clears the transmit queue. Characters can now be added to the queue using the **trace:data** command. Each new **trace:data** command adds characters to the end of the queue. When the queue contains the desired message, the transmission can be started.

Transmissions can be sent either as a single burst or at timed intervals. The **trig:seq:sour <channel> imm** command sets the Model 6065 for single burst. The **trig:seq:sour <channel> tim** command says to use timed intervals. If timed intervals are used, the time value should be set with the **trig:seq:tim <channel> <time_value>** command.

In either single burst or timed intervals, the transmission isn't started until the channel is triggered. If a channel is triggered before it has finished its previous transmission, an error is placed in the error queue and the transmission is stopped. There are three commands that can be used to trigger channels. The ***trg** command and the **trig** command both act exactly the same. They start all block mode transmit channels that are not already running with timed intervals. This means it is possible to start many channels at nearly the same time. A single channel can be started with the **trig <channel>** command.

Once a timed interval block has been triggered, it automatically receives another trigger at the end of the timed interval. This continues until the **trig:seq:tim <channel> <time_value>** command sets the time value to 0, or a ***RST** or **abort** command is received. Note that it will also stop if all the data hasn't been transmitted when it receives its next trigger. Setting the **<time_value>** to zero allows the current queue transmission to continue to the end of the queue. The ***RST** and **abort** commands will stop transmission as soon as possible.

3.4 TERMINATION CHARACTER AND TERMINATION LENGTH

When data is retrieved from the 6065, it comes back in chunks or groups of data. The 6065 needs to be told how to identify the end of those groups. The end of a group of data can be specified in three ways.

1. A particular character can be used to signal the end of the group. This can be used if every line sent to the 6065 ended with a particular character, i.e., a serial interface where every line of received data always ended with a line-feed. A user could group the data into lines by specifying a line-feed character with the **term:char 10** command (10 is the number for an ASCII line-feed character).

2. If the group of data always has the same number of characters in each group, the data group can be specified with a **term:len** command. This might be used if the received data had known fixed length character string, i.e., a peripheral that always sent 15 characters each time it reported. The user could tell the 6065 to group data into individual reports by specifying a **term:leng 15** command.

3. If the length of the data is not known and there is no single termination character that can be specified, the 6065 can be told to group data into a "whatever is available" type group. This is done with a **term:leng 0** command. Any data in the requested receive queue is returned when there is a request for data.

The end of data group indicator can also affect the format of retrieved data. If the termination length has been set to **0** (any characters available) and the format of the requested channel is set to **PACKed**, data is returned as an indefinite length block. If the data format is **PACKed** and the termination length is set to a positive number (a fixed record length), data is returned as a definite length block.

3.5 ERROR MESSAGES

When errors are detected by the module, an error message is placed in the error queue and one of the bits is set in the standard ESR. The error queue has a fixed length of two in the 6065. If more than a maximum number of error messages are placed in the error queue without being read, the last error message is over written with the queue overflow error message. The SYST:ERR? query is used to read the error queue. If there are no errors in the queue, the **No error** message is returned. See *SCPI Volume 2: Command Reference* Section 19.7 for a detailed description of error processing.

Error messages in SCPI have a common format:

<error number>,"<description>[;<device dependent info>]"

The <error number> is negative for all errors and zero for the no error condition.

0, "No error"
The error queue is completely empty. Every error message has been read or the queue was purposely cleared by *CLS. The queue is empty at power-on.

Error numbers in the range of -199 to -100 indicate a syntax error.

-100, "Command error; Line too long, scan aborted"
The input buffer is big enough to hold 4095 characters. If more than 4095 characters are sent without a terminator, the input buffer overflows and this error is generated.

-101, "Invalid character; A comma was expected but not found"
Transmit characters may be entered as a list of numbers separated by commas. A list of transmit characters was partially recognized, but some problem was encountered. Typically, this results from a misplaced or missing comma in the list.

-101, "Invalid character; A number or block was expected but not found"
A command that required either a number (data) or a block didn't find what it was looking for. This may be due to a missing parameter or an invalid block.

-102, "Syntax error; Unknown command: ..."
The module didn't recognize the command. Up to 40 characters of the offending command are included in the error message (the characters are put in where the ... appears in the error message, the ... does not appear).

-120, "Numeric data error; Buffers must have a size of at least 2"
All buffers have a minimum size of 2 characters. A command attempted to set a receive or transmit buffer to a size less than 2.

-120, 'Numeric data error; Data values are 0 to 255"
Transmit characters may be entered as a list of numbers separated by commas. Since the largest number of data bits is 8, the largest data value allowed is 255. A command to load a transmit buffer tried to enter a number outside the range of 0 to 255.

-120, 'Numeric data error; Invalid baud rate"
The baud rate of a channel can be programmed, but there are restrictions on what baud rates are allowed. An invalid baud rate was entered in a baud rate command. See the specific baud rate command for a list of valid baud rates.

-120, 'Numeric data error; Invalid number of bits"
The number of data bits for a channel is programmable, but there are restrictions on the number of bits. An invalid number of data bits was entered in a data bits command. See the specific data bits command for a list of valid data bits.

-120, 'Numeric data error; Invalid number of stop bits"
The number of stop bits for a channel is programmable, but there are restrictions on the number of bits. An invalid number of stop bits was specified in a stop bits command. See the specific stop bits command for list of valid stop bits.

-120, 'Numeric data error; Threshold must be a positive number"
The start and stop thresholds for receive pacing for a channel are programmable, but the values must be positive and inside the buffer. A negative number was specified for a pacing threshold. See the specific threshold command for the valid range.

-120, 'Numeric data error; Valid channel numbers are 1 to 4"
On a module with 4 channels, an attempt was made to specify a channel other than 1 to 4.

-120, 'Numeric data error; Valid channel numbers are 1 to 8"
On a module with 8 channels, an attempt was made to specify a channel other than 1 to 8.

-120, 'Numeric data error; Valid interfaces are 232, 422, 423 or 485"
An attempt was made to specify an interface standard other than those allowed.

-120, 'Numeric data error; Valid receive trace names are RCH1 to RCH4"
On a module with 4 channels, an attempt was made to specify a receive trace buffer other than 1 to 4.

-120, 'Numeric data error; Valid receive trace names are RCH1 to RCH8"
On a module with 8 channels, an attempt was made to specify a receive trace buffer other than 1 to 8.

-120, 'Numeric data error; Valid SAV/RCL records are 1 to 16"
An attempt was made to access a non-vol record other than 1 to 16.

-120, "Numeric data error; Valid termination lengths are 0 or larger"
A negative number was specified for the termination length of a receive buffer.

-120, "Numeric data error; Valid termination numbers are 0 to 255"
A termination character other than 0 to 255 was specified.

-120, "Numeric data error; Valid time values are 0 to 2147 seconds"
A number outside the range of 0 to 2147 was specified for a time value.

-120, "Numeric data error; Valid transmit trace names are TCH1 to TCH4"
On a module with 4 channels, an attempt was made to specify a transmit trace buffer other than 1 to 4.

-120, "Numeric data error; Valid transmit trace names are TCH1 to TCH8"
On a module with 8 channels, an attempt was made to specify a transmit trace buffer other than 1 to 8.

-121, "Invalid character in number"
A non-numeric character was encountered in what should have been a number. A missing semicolon can be one cause of this error message.

-160, "Block data error; Block length was non-numeric"
A definite length block was specified, but the length of the block was not a recognized number. Check the format of the definite length block.

-160, "Block data error; Character after # wasn't a digit"
A block was specified, but the character after the start of block character (#) wasn't a decimal digit. Check the format of the block.

-161, "Invalid block data; Expected more data than what was supplied"
When a definite length block is specified, there is a fixed length. A definite length block was specified, but the **END** indicator was recognized before all the data was found. Check the format of the definite length block.

-200, "Execution error; Can't fill buffer while using it"
This error occurs when a channel is used in block mode. When a block mode channel is transmitting, its buffer can't be filled. This error indicates there was an attempt to fill a transmit buffer while it was transmitting.

-210, "Trigger error; A block was triggered before send was finished"
This error occurs when a channel is used in block mode using timed transmit. The module attempts to send a channel buffer within the time period specified. If the time to start a new transmission occurs and the previous transmission hasn't finished, this error will occur. Some possible problems are too much data to send in the specified time interval and/or handshaking is slowing down the transmission.

-221, "Settings conflict; Not enough memory to allocate buffer"

There was an attempt to allocate more memory to buffers than is actually available. Memory is allocated with each POINts command. This error can occur when allocating non-symmetric buffers. Always allocate the smaller buffers first, then the larger buffers.

-221, "Settings conflict; RTS mode can't be set in 485"

In RS-485 mode, the module internally uses the RTS signal to control the enabling and disabling of the transmit drivers. The RTS signal can't be used in RS-485 mode.

-221, "Settings conflict; Start threshold wasn't inside buffer"

The start and stop thresholds for receive pacing for a channel are programmable, but the values must be positive and inside the buffer by a specific amount. Something was changed that made a start threshold larger than the allowed range. See the specific threshold command for the valid range.

-221, "Settings conflict; Stop threshold wasn't inside buffer"

The start and stop thresholds for receive pacing for a channel are programmable, but the values must be positive and inside the buffer by a specific amount. Something was changed that made a stop threshold larger than the allowed range. See the specific threshold command for the valid range.

-221, "Settings conflict; Termination length wasn't less than buffer size"

When using termination length to establish the end of a transmission, the length must be less than the size of the receive buffer for the specified channel. Something was changed that made the termination length larger than the allowed range.

-222, "Data out of range; Start threshold wasn't inside buffer"

The start and stop thresholds for receive pacing for a channel are programmable, but the values must be positive and inside the buffer by a specific amount. A command specified a start threshold larger than the allowed range. See the specific threshold command for the valid range.

-222, "Data out of range; Stop threshold wasn't inside buffer"

The start and stop thresholds for receive pacing for a channel are programmable, but the values must be positive and inside the buffer by a specific amount. A command specified a stop threshold larger than the allowed range. See the specific threshold command for the valid range.

-222, "Data out of range; Termination length wasn't less than buffer size"

When using termination length to establish the end of a transmission, the length must be less than the size of the receive buffer for the specified channel. A command specified a termination length larger than the allowed range.

-223, "Too much data; Transmit buffer full"

An attempt was made to put too much data into a transmit buffer. Either too much data is being put in the buffer or the data is not being sent fast enough and the buffer overflowed.

-231, "Data questionable; Framing error occurred on channel *x*"

The value *x* is a channel number from 1 to 4 on a 4-channel module and from 1 to 8 on an 8-channel module. A framing error occurred on the specified receive channel. Framing errors are usually an indication of different baud rates between the transmitting source and the receiver.

-231, "Data questionable; Overrun error occurred on channel *x*"

The value *x* is a channel number from 1 to 4 on a 4-channel module and from 1 to 8 on an 8-channel module. An overrun error occurred on the specified receive channel. Overrun errors are usually an indication of the module being too busy to service all need functions.

-231, "Data questionable; Parity error occurred on channel *x*"

The value *x* is a channel number from 1 to 4 on a 4-channel module and from 1 to 8 on an 8-channel module. A parity error occurred on the specified receive channel. Parity errors are usually an indication of different parity or data length between the transmitting source and the receiver.

-231, "Data questionable; Receive buffer overflow occurred on channel *x*"

The value *x* is a channel number from 1 to 4 on a 4-channel module and from 1 to 8 on an 8-channel module. An overflow error occurred on the specified receive channel. Overflow errors indicate the 6065's master didn't read the receive data fast enough and an input buffer overflowed.

-350, "Queue overflow"

More errors occurred than the error queue could hold. The error queue is two messages deep. If a third error is detected, the second and third messages are thrown away and this message is entered in their place.

3.6 CONNECTION DIAGRAMS

Following are the pinouts for each of the available Serial Bus standards available on the Model 6065.

3.6.1 RS-232 Configuration

First Channel as EIA/TIA-232-E Data Terminal Equipment (DTE)

	SC6065	DB25 Connector	
GND	1	7	Signal Common
	2		N/C
TXDA	3	2	Transmitted Data
	4		N/C
RXDA	5	3	Received Data
RTSA	6	4	Request to Send
CTSA	7	5	Clear to Send
DTRA	8	20	Data Terminal Ready
DSRA	9	6	Data Set Ready

Second Channel as EIA/TIA-232-E Data Terminal Equipment (DTE)

	SC6065	DB25 Connector	
GND	10	7	Signal Common
	11		N/C
TXDB	12	2	Transmitted Data
	13		N/C
RXDB	14	3	Received Data
RTSB	15	4	Request to Send
CTSB	16	5	Clear to Send
DTRB	17	20	Data Terminal Ready
DSRB	18	6	Data Set Ready

3.6.2 RS-422 Configuration

First Channel as EIA-422-A Data Terminal Equipment (DTE)

	SC6065	EIA-422-A
GND	1	Signal Common
TXDA+	2	TX+
TXDA-	3	TX-
RXDA+	4	RX+
RXDA-	5	RX-
RTSA	6	Request to Send (Out) (RS-423 levels)
CTSA	7	Clear to Send (In) (RS-423 levels)
DTRA	8	Data Terminal Ready (Out) (RS-423 levels)
DSRA	9	Data Set Ready (In) (RS-423 levels)

Second Channel as EIA-422-A Data Terminal Equipment (DTE)

	SC6065	EIA-422-A
GND	10	Signal Common
TXDB+	11	TX+
TXDB-	12	TX-
RXDB+	13	RX+
RXDB-	14	RX-
RTSB	15	Request to Send (Out) (RS-423 levels)
CTSB	16	Clear to Send (In) (RS-423 levels)
DTRB	17	Data Terminal Ready (Out) (RS-423 levels)
DSRB	18	Data Set Ready (In) (RS-423 levels)

3.6.3 RS-423 Configuration

First Channel as RS-423-A Data Terminal Equipment (DTE)

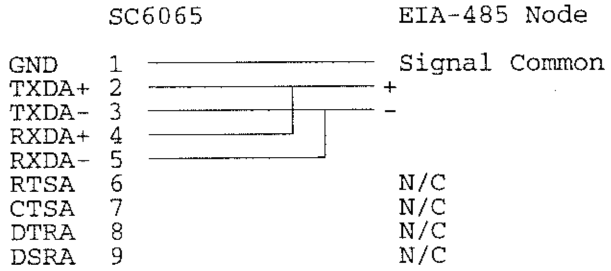
	SC6065	RS-423-A
GND	1	Signal Common
	2	N/C
TXDA	3	Transmitted Data
	4	N/C
RXDA	5	Received Data
RTSA	6	Request to Send
CTSA	7	Clear to Send
DTRA	8	Data Terminal Ready
DSRA	9	Data Set Ready

Second Channel as RS-423-A Data Terminal Equipment (DTE)

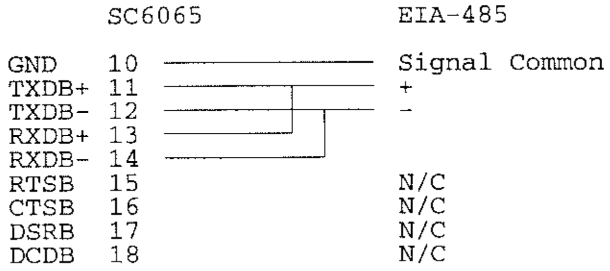
	SC6065	RS-423-A
GND	10	Signal Common
	11	N/C
TXDB	12	Transmitted Data
	13	N/C
RXDB	14	Received Data
RTSB	15	Request to Send
CTSB	16	Clear to Send
DTRB	17	Data Terminal Ready
DSRB	18	Data Set Ready

3.6.4 RS-485 Configuration

First Channel as EIA-485 Node



Second Channel as EIA-485 Node



4.1 GENERAL DESCRIPTION

The Model 6065 VXIbus Serial Interface Module is available as a 4-Channel or 8-Channel unit depending upon whether the second four channels are installed on the main logic board of the module. The Serial Interface Module consists of a main logic board and a VXIbus Message-Based Interface daughter card. The main logic board contains:

- Serial UARTs (Universal Asynchronous Receiver/Transmitter)
- RS-232, RS-422, RS-423 and RS-485 drivers and receivers
- Self-test loop back relays and buffer RAMs

The VXIbus Message-Based Interface daughter card contains:

- VXIbus Registers
- 68000 CPU
- EPROM containing the module's program
- EEPROM for storing configuration data
- System RAM
- System timer

4.2 VXIBUS MESSAGE-BASED INTERFACE

4.2.1 General

The VXIbus Message-Based Interface implements a complete interface per the VXIbus Specification Revisions 1.3 and 1.4. The daughter card contains all the VXIbus registers implemented in dual-ported memory, one 68000 CPU running at 8 MHz for the 4-Channel Module or at 12.5 MHz for the 8-Channel Module. The interface daughter card also includes 128k bytes of program memory, 16k bytes of EEPROM, 64k bytes of system RAM, one MC68230 Timer/Counter and a Local Bus Interface (which is not used in the Model 6065).

4.2.2 Microprocessor Clock

The microprocessor clock is derived from two possible sources depending upon whether the interface is running at 8 MHz or at 12.5 MHz. If the interface is running at 8 MHz, the VXIbus system clock is buffered by U33 and routed to the CPU via jumper JP5. If the interface is running at 12.5 MHz, a 25 MHz TTL output oscillator is installed at U13, and no jumper is installed at JP5. Both clock sources are divided by two by U7 providing a clean 50% duty cycle clock for the CPU.

4.2.3 Microprocessor Memory

The Message-Based Interface uses a single 64k by 16 EPROM (U1) for its program memory, two 8k by 8 EEPROMs (U5 and U6) for non-volatile storage of configuration data, and two 32k by 8 static RAMs for system use (U2 and U3). Decoding for the memories, along with the other peripherals on the daughter card, is implemented by PAL (programmable array logic) U12. U12 also handles the logic required for the DTACK and VPA functions. U4 provides the necessary logic to control the write functions for the RAMs and EEPROMs allowing byte write operations to occur.

4.2.4 VXIbus Interface

The VXIbus is connected to the interface daughter card via P101 and P102. The data bus is buffered by U37 and U38 which are octal bi-directional buffer/latches configured to buffer the data coming into the card and latch the data going out, depending upon the type of bus cycle (read or write). The address modifier lines are decoded by U34 and U35, and provide an enable to U26 when an A16 access is occurring on the VXIbus in the upper quadrant (#HC000 through #HFFFF). The specific board address is determined by the setting of SWl, and is compared by U26 to produce the signal VREGENA! indicating this specific interface is being accessed.

Dynamic Configuration of the interface's address is provided by U41 and U31. The configured offset address is written into U41 and, if it is output enabled, it will over-drive the SW1 setting. Dynamic Configuration is enabled by setting SW1 to all ones (#HFF). This is seen by U42, and the FF! signal enables the offset to be written into U41 via U39 and U40.

In order for the CPU to have knowledge of its address, U27 provides a method of reading the address setting.

The principal state machine which controls VXIbus transfers is contained in PAL U36. It provides timing for the data buffers U37 and U38 as well as write and enable signals to the dual ported RAMs, U28 and U29, which implement the VXI registers.

When a VXIbus register is written to, U36 provides the enables (REGCE0! and REGCE1!) and the write strobe (VWRITE!) to U28 and U29. U37 and U38 are also enabled to pass the data through to the dual-ported RAMs (signal IEVDB!). When a VXIbusregister is read, U36 enables U28 and U29, and latches their output into U37 and U38 with the DCLK signal. U37 and U38 are output enabled to the VXIbus by the OEVDB0 and OEVDBl signals which are controlled by the VDS0! and VDSl! signals respectively.

When the dual-ported RAMs are accessed, the address of the register (which was accessed) is latched by U30 and U31. U11 prevents further accesses to the interface until the CPU can read the register which has changed. This process produces an interrupt on IRQ6! which causes the event to be serviced and the interface to be re-enabled.

4.2.5 Microprocessor Interrupts

The Message-Based Interface MC68000 is configured for auto-vectored interrupts. This allows for seven unique interrupt levels. The interrupts are encoded by U14. The highest level interrupt is NMI! and is not used. IRQ6 is used by the VXIbus interface. IRQ5! is used by the VXIbus trigger inputs. IRQ4! is available to the user circuitry and is brought to P100. IRQ3! is available to the local bus. IRQ2! is also available to the user circuitry, and is also brought to P100 and the lowest priority interrupt. IRQ1! is used by the MC68230 timer/counter.

4.2.6 Timer/Counter

The Message-Based Interface has an MC68230 Timer/Counter Ul9 which provides a timing function and some parallel I/O. It is used to provide the real time clock function of the interface and control of the VXIbus TTL trigger lines. The clock source for the MC68230 is the same as the microprocessor clock divided by 2 by U7.

The parallel I/O provides the source for the Dynamic Configuration enable signal DCENA!, the trigger source from the timer function, and the TTL trigger bus.

Signals from port PA0 through PA7 set the trigger output line, the trigger input line and the trigger input and output enables. These control lines are decoded by U16 for the output function and U15 for the input function. U21 buffers the trigger outputs and provides the required open collector drive for these lines. The selected trigger input may cause an interrupt to the CPU as well as being made available to the user connector P100.

Signals TROUT! enable the TTL trigger output driver; SYSFAIL causes the VSYSFAIL! to go true; FAILLED causes the SYSFAIL LED to light.

The test point El provides the CPU with the running speed. If a jumper is installed between pins 3 and 4, the interface is expected to run at 8 MHz. If it is left out, it is expected to run at 12.5 MHz. This affects the timer function. In addition, El pins, 1 and 2, are used to clear out the non-volatile memory. If pins 1 and 2 are shorted at power-up, the default values are written into the EEPROMs, and provide a method to restore the interface to the original configuration.

4.2.7 VXIbus Interrupts

Signals from the Ul9 ports PB0, PBl and PB2 set the VXIbus interrupt level, and PB4 reads to see if an interrupt is pending. The setting of IRQSEL0 through 3 determines which interrupt level to drive, and which acknowledge response. U17 decodes the three lines into the seven IRQ lines on the back plane which are buffered by the open collector driver U22. The IRQSEL0 through 3 also sets the compare value for U18, U8 and U24 to respond to during the interrupt acknowledge cycle. This cycle drives the VIACKOUT! line when a VIACKIN! signal and matching address are found. Ull and U20 provide synchronization of VXIbus interrupts so none are lost due to the asynchronous nature of an interrupt event.

The VXIbus interrupt is started by clocking the SETVXIRQ! signal and may be cleared, if necessary, by clocking the CLRVXIRQ! signal.

4.2.8 SYSFAIL and Access Indicators

The Message-Based Interface provides an access LED to indicate the interface is being communicated with. When U36 runs through an access cycle, it sends a signal ALTDTACK! to U25 which stretches the length of the pulse to about 100 mS in length. U25 then drives Q1 which in turn lights the LED.

The SYSFAIL LED is driven by Q2 which is driven by U19 under program control. It indicates the interface has not passed its self-test successfully.

4.2.9 Local Bus Interface

The Model 6065 does not use the local bus interface feature of the Message-Based Interface.

4.3 MAIN LOGIC BOARD

4.3.1 General

The serial interface is implemented using dual UART. Each dual UART implements two channels of serial interface. The 4-channel module has two of these integrated circuits while the 8-channel module has four. To describe the operation of the Model 6065 Serial Interface Module, refer to channels 1 and 2 only. Note that this information applies equally to the other channels.

4.3.2 Microprocessor Bus

The microprocessor bus is brought to the main logic board via P7. U50, U53 and U56 provide address bus buffering on the main logic board to reduce loading on the 68000 CPU. U59 and U62 provide data bus buffering also to reduce CPU loading. U65 is a PAL (programmable array logic) device which implements the necessary control logic for the main logic board to transfer data to and from the VXIbus Message-Based Interface daughter card. It provides control signals to the data buffers, enabling them and setting their data direction. It provides a decoded enable signal for the devices on the main logic board (signal named PORTENA* (Note that any signal name which ends in a "*" is a low true signal). It provides enables for the buffer RAMs (named RAM0ENA* and RAM1ENA*) and a DTACK signal to the 68000 CPU (named XDTACK*).

The PORTENA* signal is further decoded by U68 and U69 which provide the ultimate decoded enables for the eight output ports (signals named PORT0* through PORT7*), four dual UARTs (signals named DUART0* through DUART3*), an unused read back port (named READPORT*), and an unused enable named FPGACE*.

The output ports are implemented using 74HCT273 octal latches with resets. The latches are cleared during power-up reset (by the signal named RESET*), and can be written to thereafter by the CPU. Six of the output ports (U51, U54, U57, U60, U63, U66) are used to set the self-test configuration relays. They drive the ULN-2803A relay driver ICs (U52, U55, U58, U61, U64 and U67 respectively). These relay drivers in turn control the relays (Kl through K48) on the main logic board. Note that the four channel Model 6065 only requires K1 through K24 and their respective drivers and latches.

4.3.3 Serial UARTs

The dual UARTs (U2, U3, U4 and U5) are enabled with the DUART0* through DUART3* signals, and use the buffered address and data bus to communicate with the 68000 CPU. The dual UART requires separate read and write control signals (named READ* and WRITE*) which are generated by PAL U78. The timing for these signals is derived from the R/W*, AS* and PORTENA* signals. R26 and C65 provide termination for the WRITE* signal, while R27 and C66 provide termination for the READ* signal.

The dual UARTs require an external source to set their baud rate. Ul provides a 3.6864 MHz clock (named BAUDCLK) to the UARTs. C1 and R4 provide a termination for this clock signal to prevent excessive ringing.

Each dual UART implements all the necessary logic to provide two bi-directional serial communications channels. They provide two transmit outputs, two receive inputs, two RTS (Request To Send) outputs, two CTS (Clear To Send) inputs, five general purpose inputs and six general purpose outputs.

Two of the general purpose inputs (IP2 and IP3) are used to implement the DSR (Data Set Ready) signals, and two of the general purpose outputs are used to implement the DTR (Data Terminal Ready) signals. The remaining general purpose inputs and outputs are wired to pads for ease of customization. The unused inputs are pulled up to a high logic level to insure consistent data read back.

The serial interface signals are routed through P3 to allow the signals to be intercepted for customized serial interface hardware (such as electrically isolated interfaces or fiber optic interfaces). +5, +12 and -12 volts are also provided on P3 to support such interfaces.

In the standard Model 6065 Serial Interface Module, the signals are jumpered across P3 (pin 1 to pin 2; pin 3 to pin 4, etc.), and are then routed to the on-board drivers and receivers.

4.3.4 Line Drivers and Receivers

The Model 6065 Serial Interface Module is capable of transmitting and receiving in either RS-232, RS422, RS-423 or RS-485 compatible modes. The configuration is selectable through software control, and does not require the module be opened.

The Model 6065 uses a UA9636A for its transmit and handshake buffers in the RS-232 and RS-423 modes. It takes a TTL input signal and produces an output which swings from +6 volts to -6 volts, which is consistent with the requirements of these two standards. When RS-422 or RS-485 is required, the Model 6065 uses a 26LS3lC line driver for its transmit buffers. Note that in the RS-422 mode, the handshake lines are RS-423 compatible and in the RS-485 mode, handshake lines are not used.

For its receivers, the Model 6065 uses the SN75189 to support the RS-232 mode, and uses the 26LS32A for the RS-422, RS-423 and RS-485 modes. The 26LS32A input is wired differentially for the RS-422 and RS-485 modes. It is wired for single-ended input in the RS-423 mode by connecting the positive input to ground at the front panel connector and driving the negative input.

On channel 1, the UART's transmit output drives both U6 and U8. The RTS and DSR signals are also routed to U13. If RS-232 or RS-423 is selected, Kl is disabled routing U6's transmit output to the front panel connector J100. K5 and K6 are then enabled to disconnect the output of two U8 from J100. Additionally, K3 and K10 are disabled connecting U13's RTS and DSR outputs to J100. If RS-422 or RS-485 is selected, Kl is enabled, and K5 and K6 are disabled routing the output of U8 to J100.

The primary electrical difference between RS-422 and RS-485 is that the output drive should only be enabled when it is commanded to transmit data. This is achieved by using the UARTs' RTS output to control the transmit driver's enable in the RS-485 mode. This is handled by the PAL U16. When the RS-485 mode is selected, RTS enables U8's output. If RS-422 is selected, U8 is always enabled.

The Serial Interface Module's inputs (receive, CTS and DCD) are routed to either the SN75189 line receivers (U7 and Ull) for RS-232 mode), or to the 26LS32A line receivers (U9 and U14) for the RS-422, RS-423 or RS-485 modes. In the RS-232 mode, Kl, K3 and K9 are disabled, and K5, K6 and K9 are enabled routing the input signals to U7 and Ull. In the other modes, Kl, K3 and K9 are enabled while K5, K6 and K9 are disabled.

The outputs of the line receivers are selected depending on the mode selected by PAL U12. If RS-232 is selected, the outputs of U7 and Ull are selected. While in the other modes, the outputs of U9 and U14 are selected. The signal REC422A0 controls the input source and is controlled by U72.

4.3.5 Self-Test Loop Back

The Model 6065 provides for driver level loop back self-test. This ensures the drivers and receivers are fully functional.

When Kl is energized, it routes the RS-232 compatible output of U6 to the input of U7 instead of to J100. During self-test, data is transmitted in this mode and received by the same UART. The data is compared for integrity. Similarly, energizing K3 wraps RTS back to U7; K10 wraps DSR back to U14. In the RS-422 mode, K5 and K6 wrap transmit back to U9. The RS-423 and RS-485 modes are confirmed by checking both the RS-232 and RS-422 modes as the required drivers and receivers have been checked during those tests.

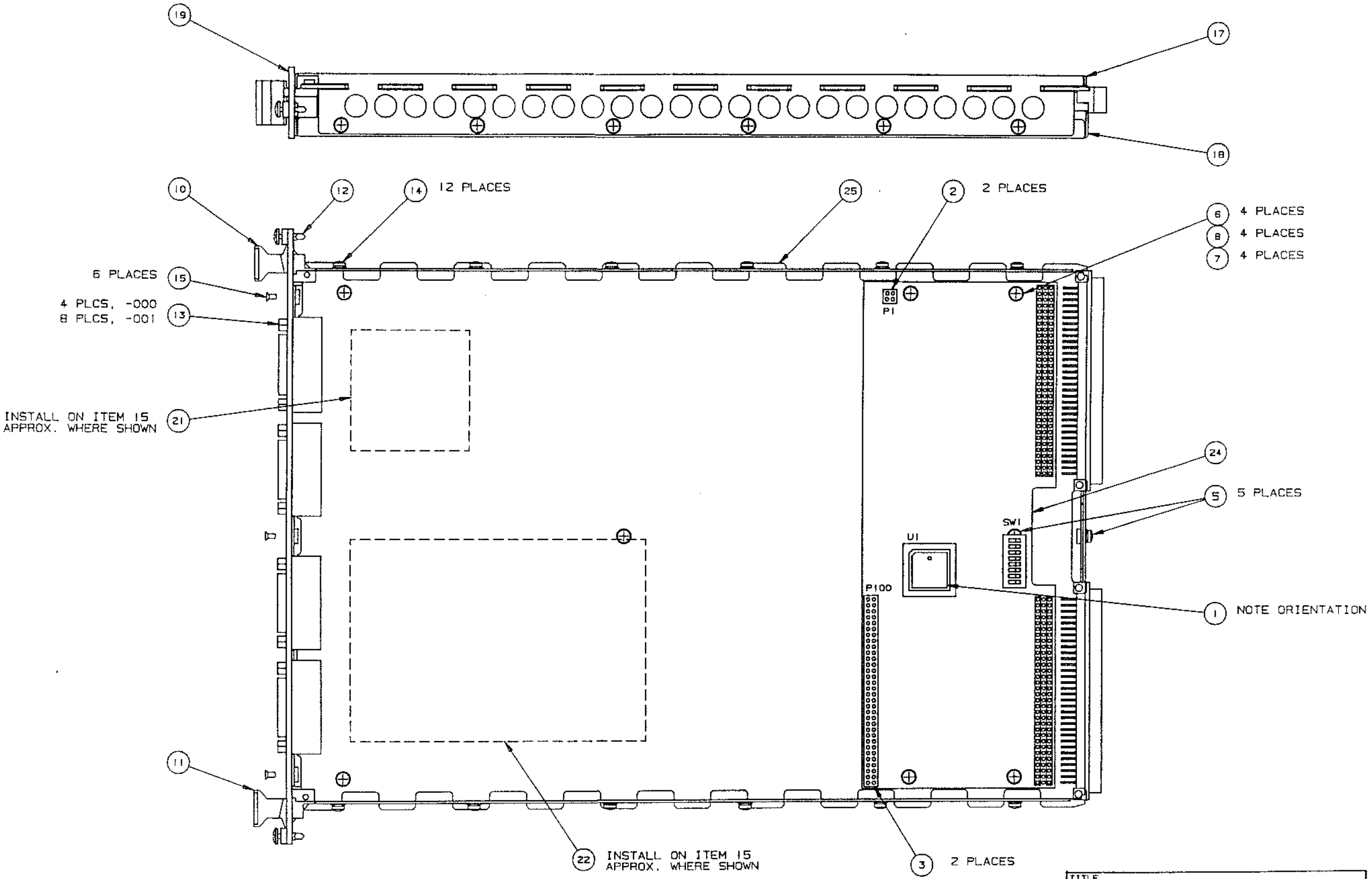
The self-test loop back procedure is run in both the RS-232 and RS-422 modes for each channel to confirm the integrity of the Serial Interface Module.

4.3.6 Buffer RAMs

The main logic board also contains the buffer RAMs which hold the transmit and received data patterns. These RAMs (U73, U74, U75 and U76) may be either 32k by 8 or 128k by 8 static RAMs providing a total of either 128k bytes or 512k bytes of total buffer memory.

U78 provides the control signals to output enable the RAMs (signals named LRAMOE* and URAMOE*), and provide the write strobes for the RAMs (signals named LRAMWRITE* and URAMWRITE*). These signals are derived from signals AS*, R/W*, LDS* and UDS*.

This section is intentionally left out.

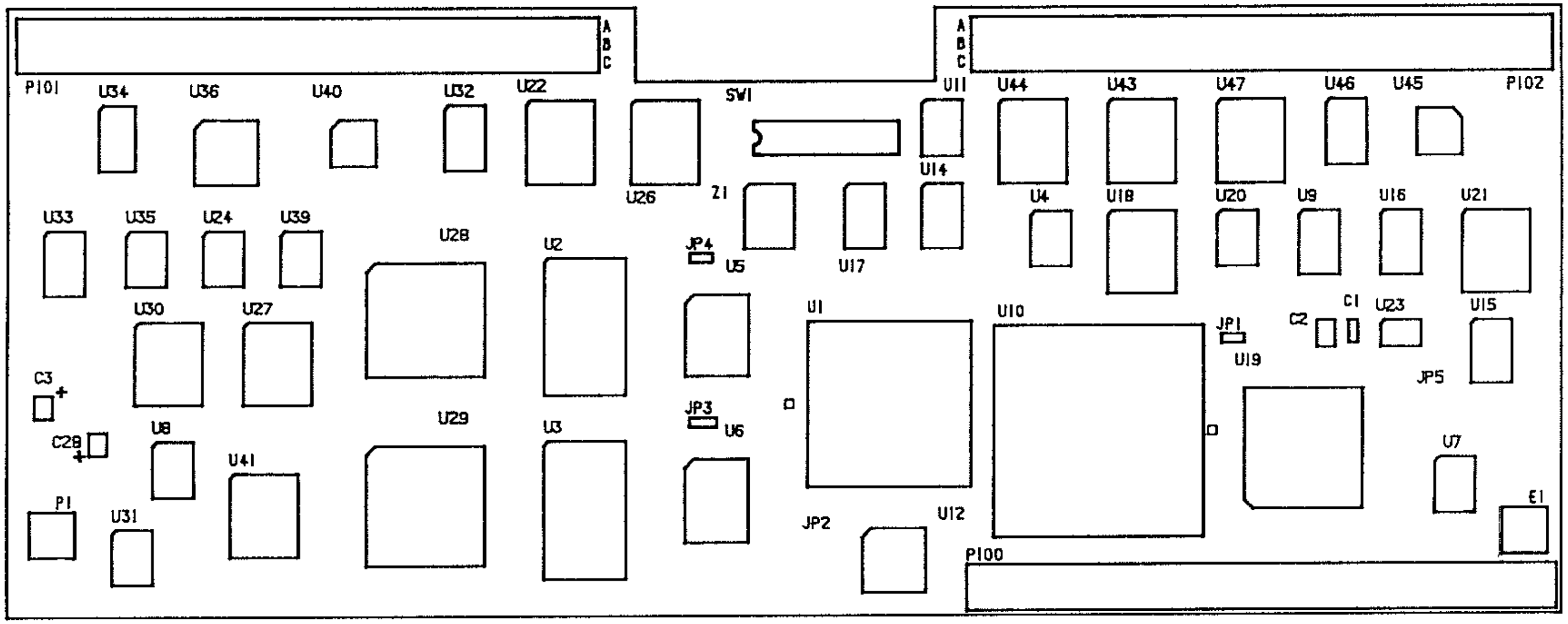


2. APPLY SERIAL NUMBER TO ITEM 18 USING INDELIBLE INK

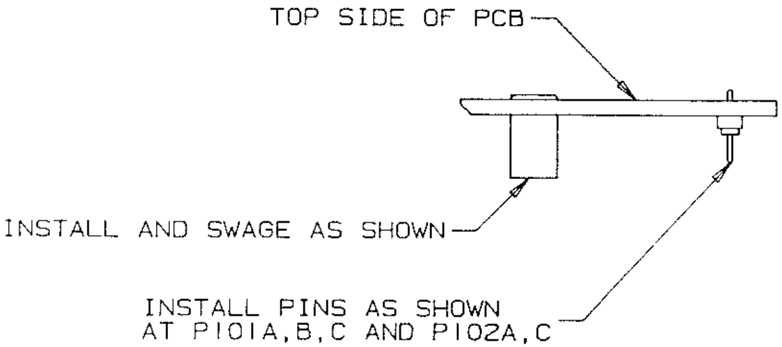
1. ASSEMBLE PER MIL-STD-454

NOTES: UNLESS OTHERWISE SPECIFIED

TITLE			
FINAL ASSY, 4/8 CHANNEL VXI SERIAL INTERFACE			
SIZE D	CAGE CODE	DWG NO. 70-0036-000/001	REV. A
SCALE NONE		SHEET 1 OF	



TOP SIDE



3. P1 AND P100 ARE INSTALLED ON THE TOP SIDE OF PCB

2. REFER TO 50-0011-000 FOR SCHEMATIC

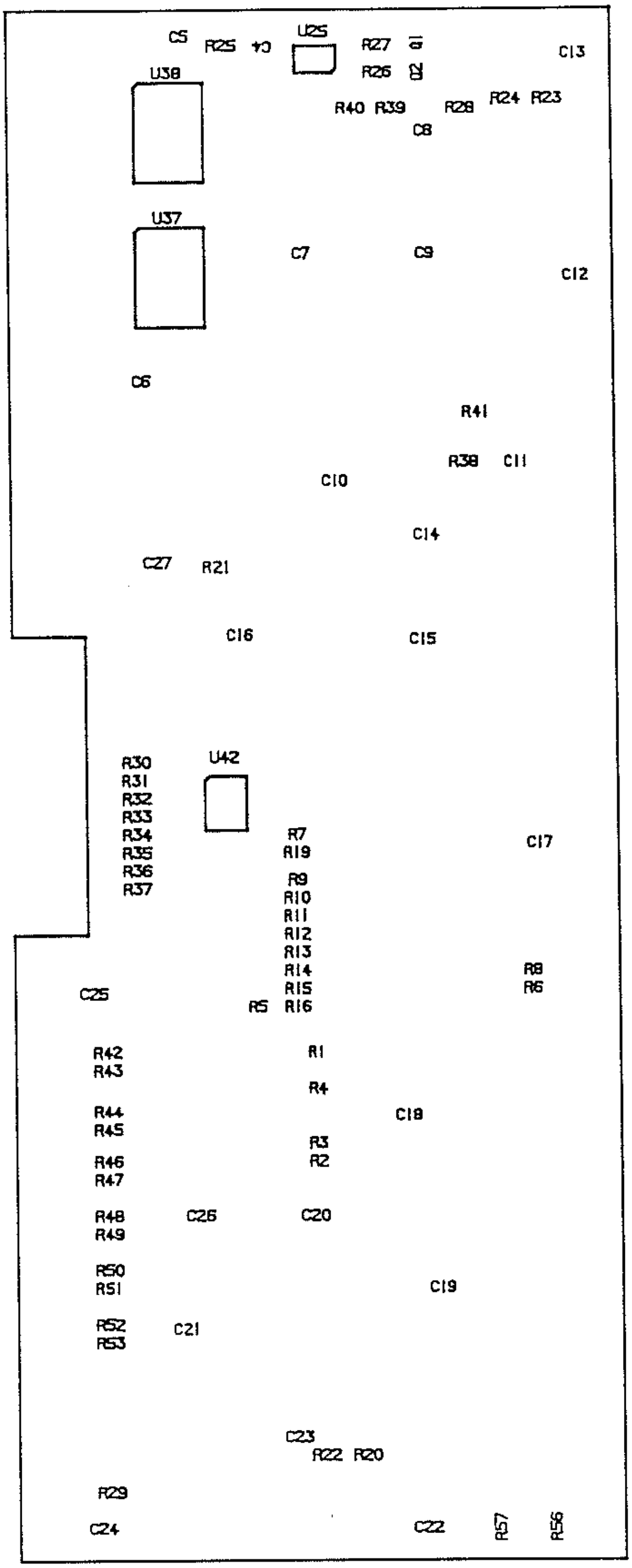
1. ASSEMBLY IS TO CONFORM TO VTI WORKMANSHIP STANDARDS

NOTES: UNLESS OTHERWISE SPECIFIED:

RACAL-DANA Instruments Inc.
IRVINE, CALIFORNIA

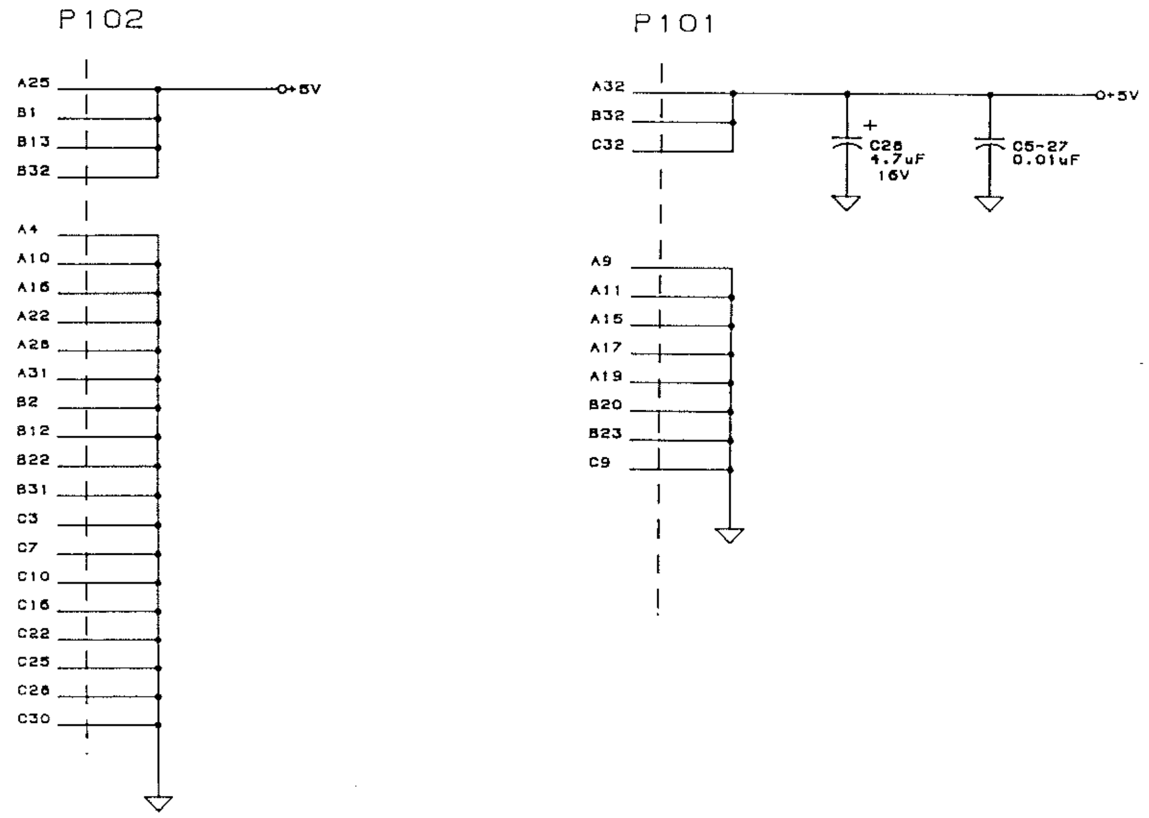
PCB ASSY., VXI
MESSAGE BASED INT

SIZE	CODE IDENT NO	DWG NO	REV
A	21793	52-0011-000	1
SCALE —		SHEET 1	OF 2



BOTTOM SIDE

SIZE	CODE IDENT NO	DWG NO	REV
A	21793	52-0011-000	1
SCALE —		SHEET 2 OF 2	



5. U13 AND JP5 ARE INSTALLED ON 12.5 MHz UNITS ONLY

4. INSTALL JP2 FOR 256K X 16 EPROM
DELETE JP2 FOR 64K X 16 AND 128K X 16 EPROM

3. INSTALL JP3 AND JP4 FOR 32K X 8 NVM
DELETE JP3 AND JP4 FOR 8K X 8 NVM

2. ALL CAPACITORS ARE 20% 50V

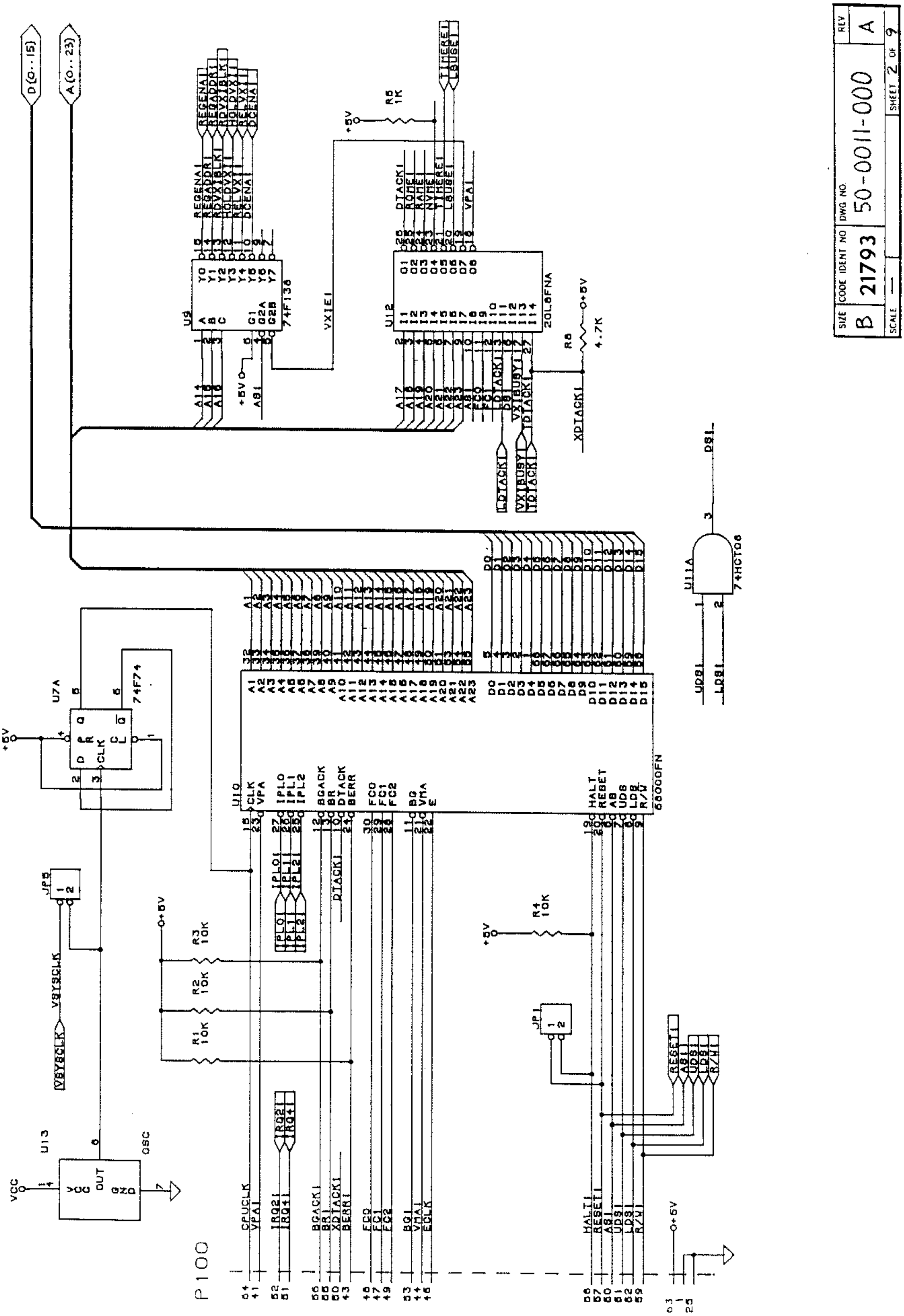
1. ALL RESISTORS ARE 5% 1/8 WATT

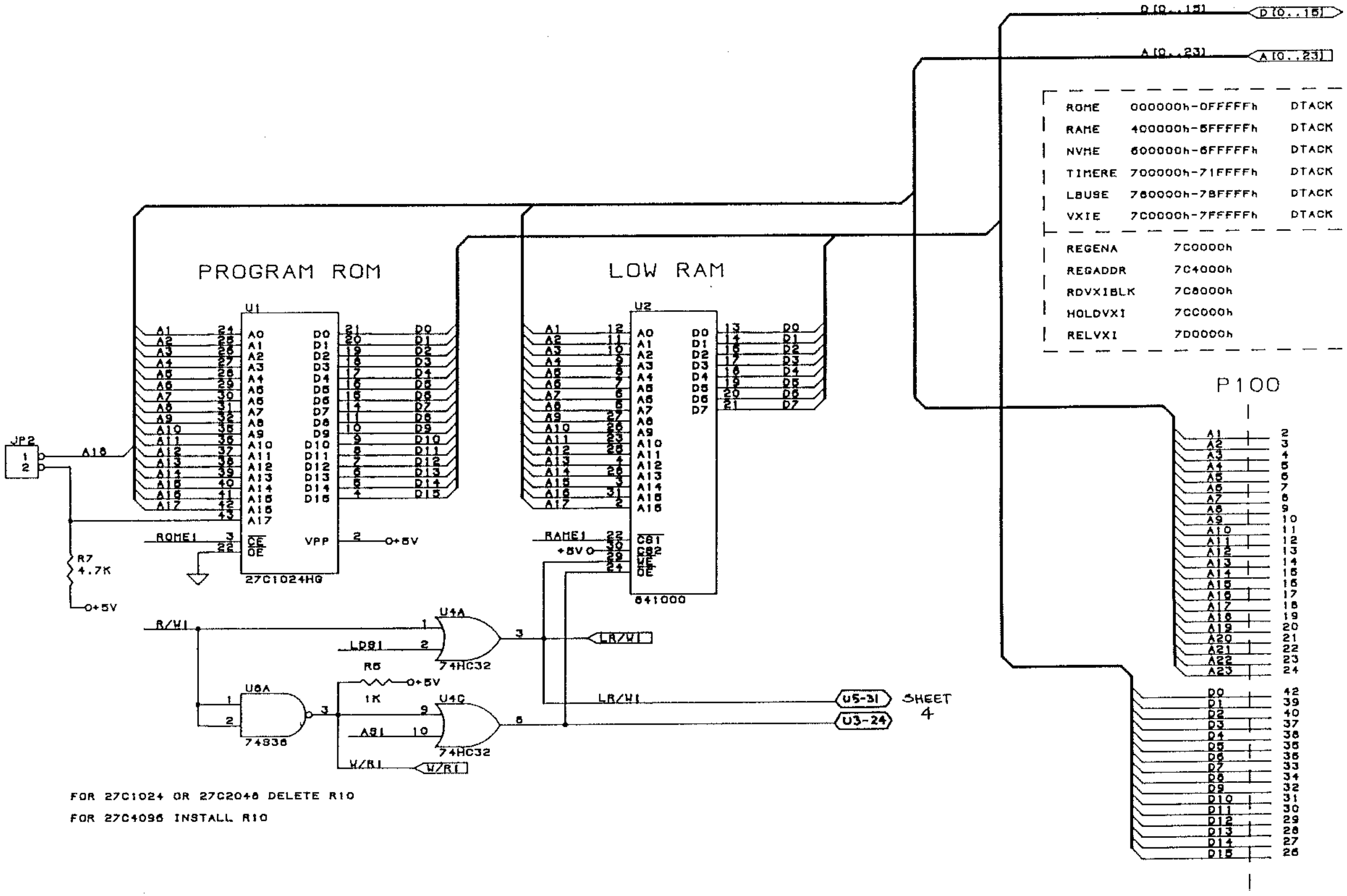
NOTES: UNLESS OTHERWISE SPECIFIED

RACAL-DANA Instruments Inc.
IRVINE, CALIFORNIA

SCHEMATIC, MESSAGE
BASED DEVICE

SIZE	CODE IDENT NO	DWG NO.	REV
B	21793	50-0011-000	A
SCALE —		SHEET 1 OF 9	


SIZE B
CODE IDENT NO 21793
DWG NO. 50-0011-000
REV A
SCALE —
SHEET 2 OF 9
P100
U13
OSC
VCC
U7A
74F74
JP5
VSYSCLK
U10
68000FN
U9
74F138
U12
20L8FNA
U11A
74HCT08
R1 10K
R2 10K
R3 10K
R4 10K
R5 1K
R6 4.7K
JP1
+5V
D(0..15)
A(0..23)
CPUCLK
VPA1
IRQ21
IRQ41
BGACK1
BR1
XDTACK1
BERR1
FC0
FC1
FC2
BG1
VMA1
ECLK
HALT1
RESET1
AS1
UDS1
LDS1
R/W1
IPL01
IPL11
IPL21
DTACK1
REGENA1
REGADDR1
RDVXIBLK1
HOLDVXI1
RELVXI1
DCENA1
VXIE1
DTACK1
ROME1
RAME1
NVME1
TIMERE1
LBUSE1
VPA1
LDTACK1
VXIBUSY1
TDTACK1
XDTACK1
DS1



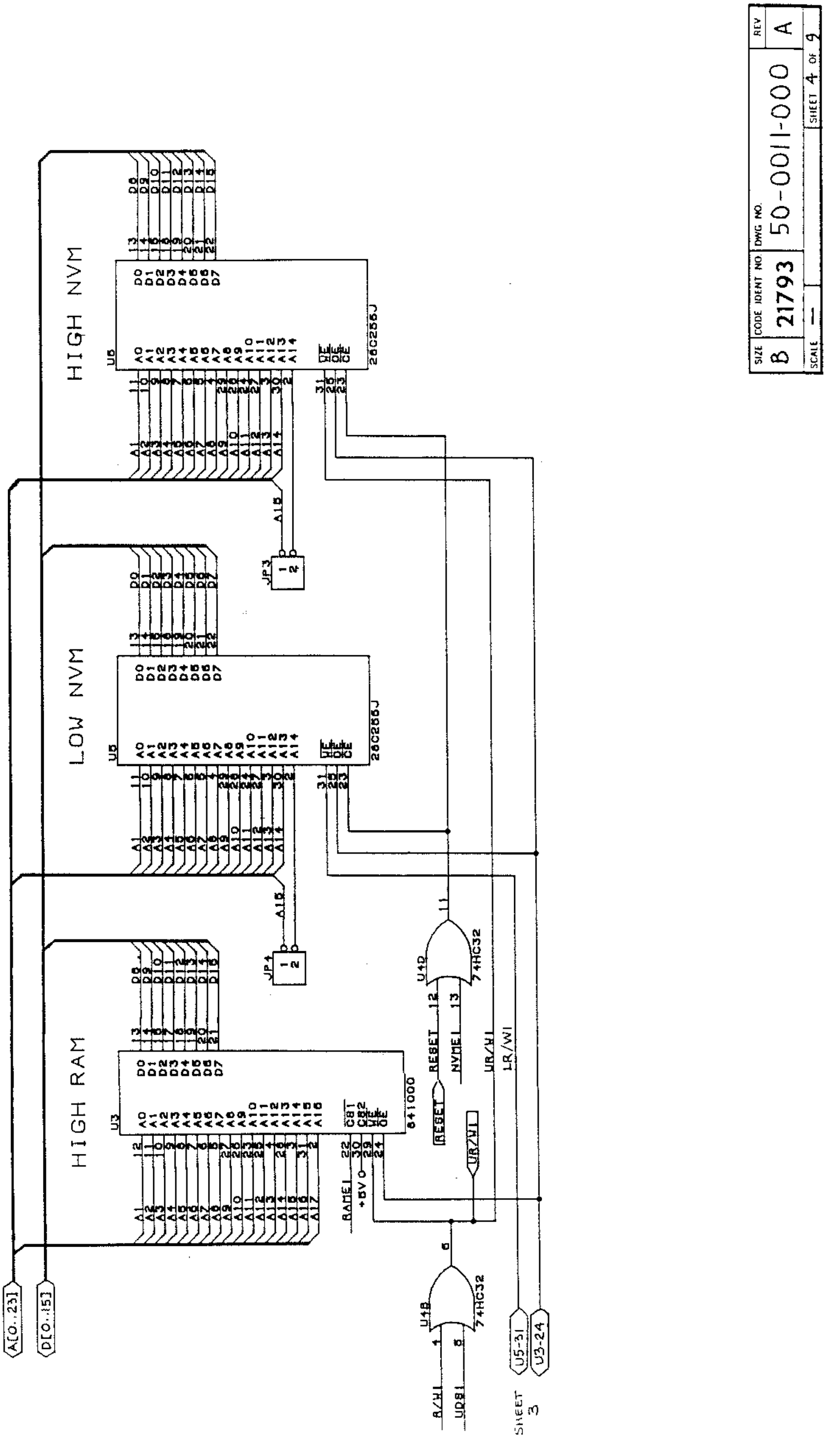
D [0..15]
A [0..23]
PROGRAM ROM
U1
27C1024HG
A0 A1 A2 A3 A4 A5 A6 A7 A8 A9 A10 A11 A12 A13 A14 A15 A16 A17
CE
OE
VPP
D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 D10 D11 D12 D13 D14 D15
ROME1
O+5V
JP2
1
2
A18
R7
4.7K
LOW RAM
U2
641000
A0 A1 A2 A3 A4 A5 A6 A7 A8 A9 A10 A11 A12 A13 A14 A15 A16
CS1
CS2
WE
OE
D0 D1 D2 D3 D4 D5 D6 D7
RAME1
+5V O
LR/W1
R/W1
LDS1
U4A
74HC32
U4C
74HC32
U5A
74S38
R5
1K
AS1
W/R1
U5-31
U3-24
SHEET 4
P100
A1 A2 A3 A4 A5 A6 A7 A8 A9 A10 A11 A12 A13 A14 A15 A16 A17 A18 A19 A20 A21 A22 A23
D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 D10 D11 D12 D13 D14 D15

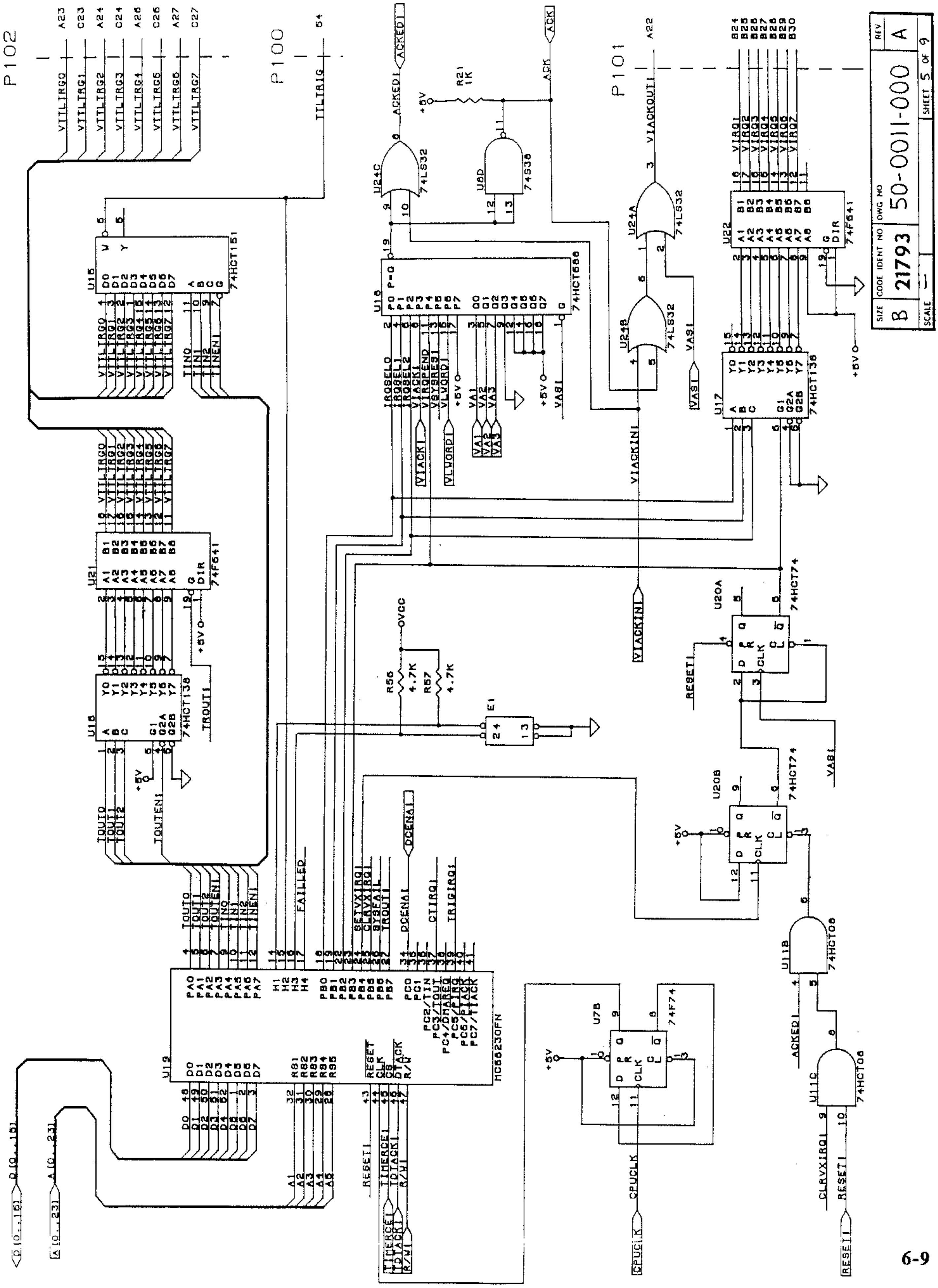
ROME	000000h-0FFFFFh	DTACK
RAME	400000h-5FFFFFh	DTACK
NVME	600000h-6FFFFFh	DTACK
TIMERE	700000h-71FFFFh	DTACK
LBUSE	780000h-78FFFFh	DTACK
VXIE	7C0000h-7FFFFFh	DTACK
REGENA	7C0000h	
REGADDR	7C4000h	
RDVX1BLK	7C8000h	
HOLDVX1	7CC000h	
RELVX1	7D0000h	

FOR 27C1024 OR 27C2048 DELETE R10

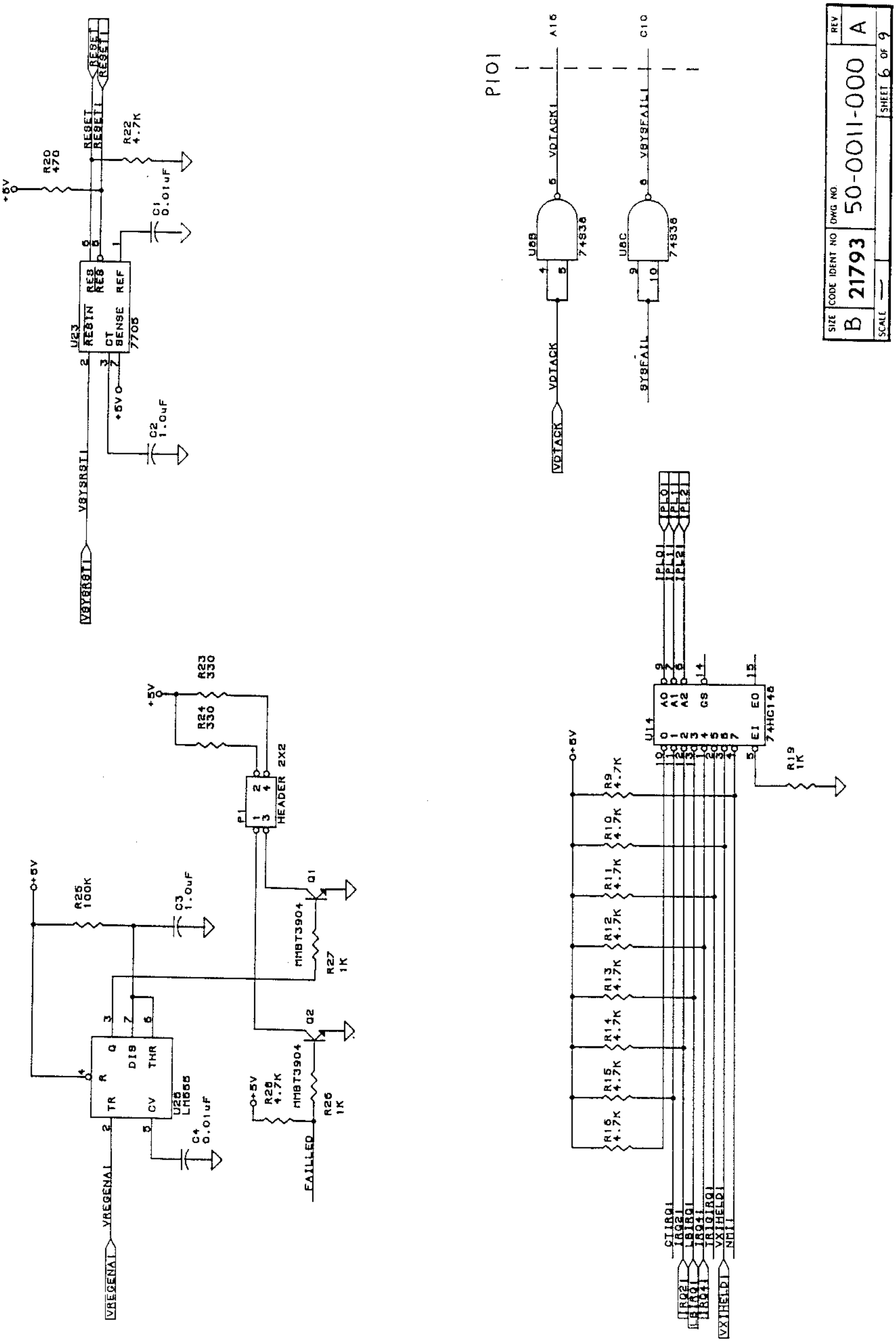
FOR 27C4096 INSTALL R10

SIZE	CODE IDENT NO	DWG NO.	REV
B	21793	50-0011-000	A
SCALE —		SHEET 3 OF 9	

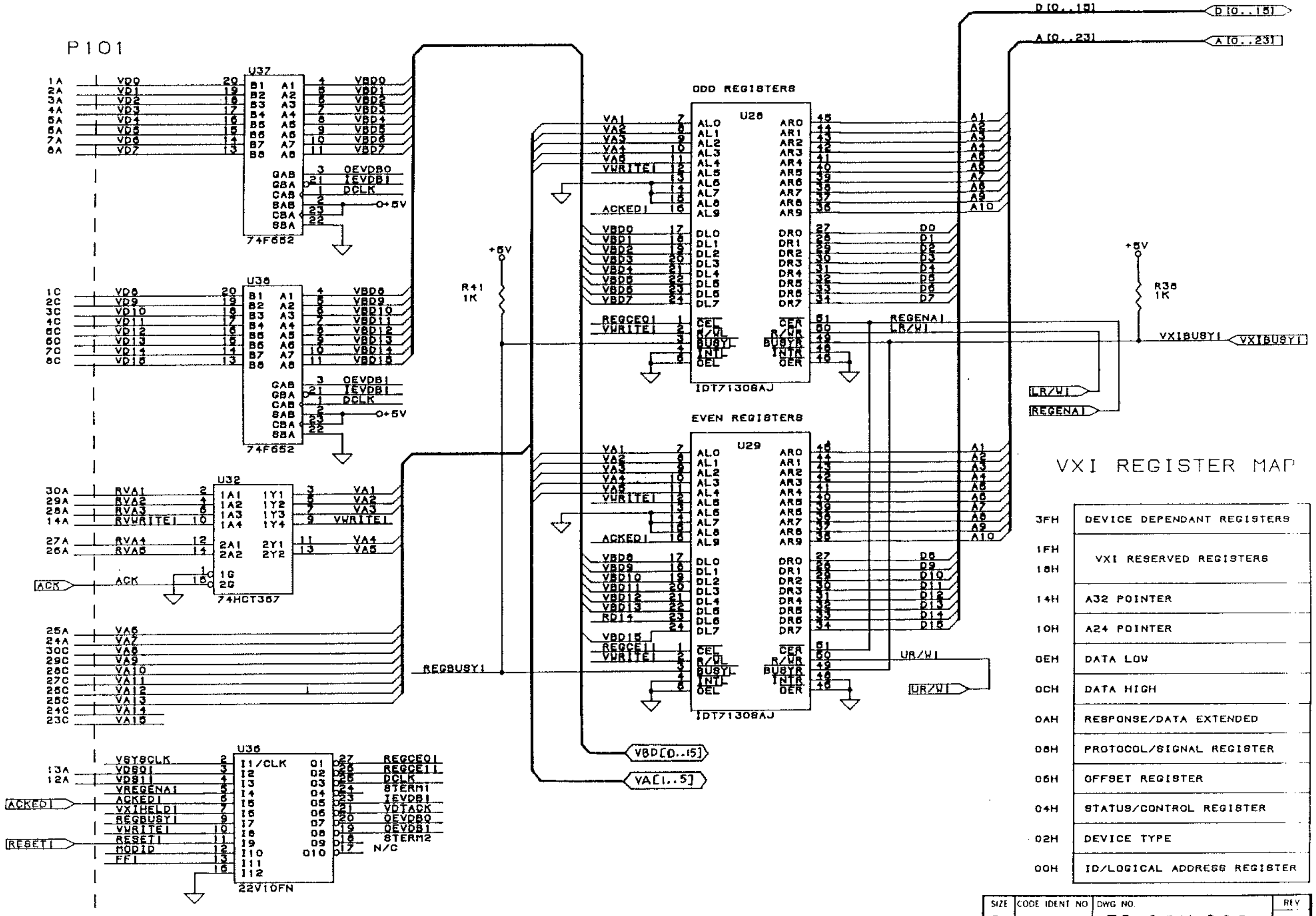

HIGH NVM
LOW NVM
HIGH RAM
U5
U5
U3
28C256J
28C256J
841000
D0 D1 D2 D3 D4 D5 D6 D7
A0 A1 A2 A3 A4 A5 A6 A7 A8 A9 A10 A11 A12 A13 A14
A15 A16
WE OE CE
CS1 CS2 WE OE
D8 D9 D10 D11 D12 D13 D14 D15
A1 A2 A3 A4 A5 A6 A7 A8 A9 A10 A11 A12 A13 A14
A15
A17
JP3
JP4
1 2
U4D
U4B
74HC32
RESET
NVME1
UR/W1
LR/W1
RAME1
+5V
A[0..23]
D[0..15]
U5-31
U3-24
SHEET 3
R/W1
UDB1
SIZE
B
CODE IDENT NO
21793
DWG NO
50-0011-000
REV
A
SCALE
—
SHEET 4 OF 9



P102
P100
P101
VTTLTRG0
VTTLTRG1
VTTLTRG2
VTTLTRG3
VTTLTRG4
VTTLTRG5
VTTLTRG6
VTTLTRG7
A23
C23
A24
C24
A25
C25
A27
C27
TTLTRIG
54
ACKED1
ACK
R21
1K
+5V
U24C
74LS32
U8D
74S38
U24A
U24B
VIACKOUT1
VIACKIN1
A22
VAS1
VIRQ1
VIRQ2
VIRQ3
VIRQ4
VIRQ5
VIRQ6
VIRQ7
B24
B25
B26
B27
B28
B29
B30
U22
74F641
U15
74HCT151
U16
74HCT688
U17
74HCT138
U18
74HCT138
U21
74F641
IRQSEL0
IRQSEL1
IRQSEL2
VIACK1
VIRQPEND
VSYSRES1
VLWORD1
VA1
VA2
VA3
TIN0
TIN1
TIN2
TINEN1
TOUT0
TOUT1
TOUT2
TOUTEN1
TROUT1
U20A
U20B
74HCT74
RESET1
E1
R56
4.7K
R57
OVCC
FAILLED
SETVXIRQ1
CLRVXIRQ1
SYSFAIL
DCENA1
CTIRQ1
TRIGIRQ1
U19
MC68230FN
PA0 PA1 PA2 PA3 PA4 PA5 PA6 PA7
H1 H2 H3 H4
PB0 PB1 PB2 PB3 PB4 PB5 PB6 PB7
PC0 PC1 PC2/TIN PC3/TOUT PC4/DMAREQ PC5/PIRQ PC6/PIACK PC7/TIACK
D0 D1 D2 D3 D4 D5 D6 D7
RS1 RS2 RS3 RS4 RS5
RESET CLK CS DTACK R/W
D(0..15)
A(0..23)
A1 A2 A3 A4 A5
TIMERCE1
TDTACK1
R/W1
U7B
74F74
CPUCLK
U11B
U11C
74HCT08
ACKED1
CLRVXIRQ1
RESET1
SIZE B
CODE IDENT NO 21793
DWG NO 50-0011-000
REV A
SCALE —
SHEET 5 OF 9



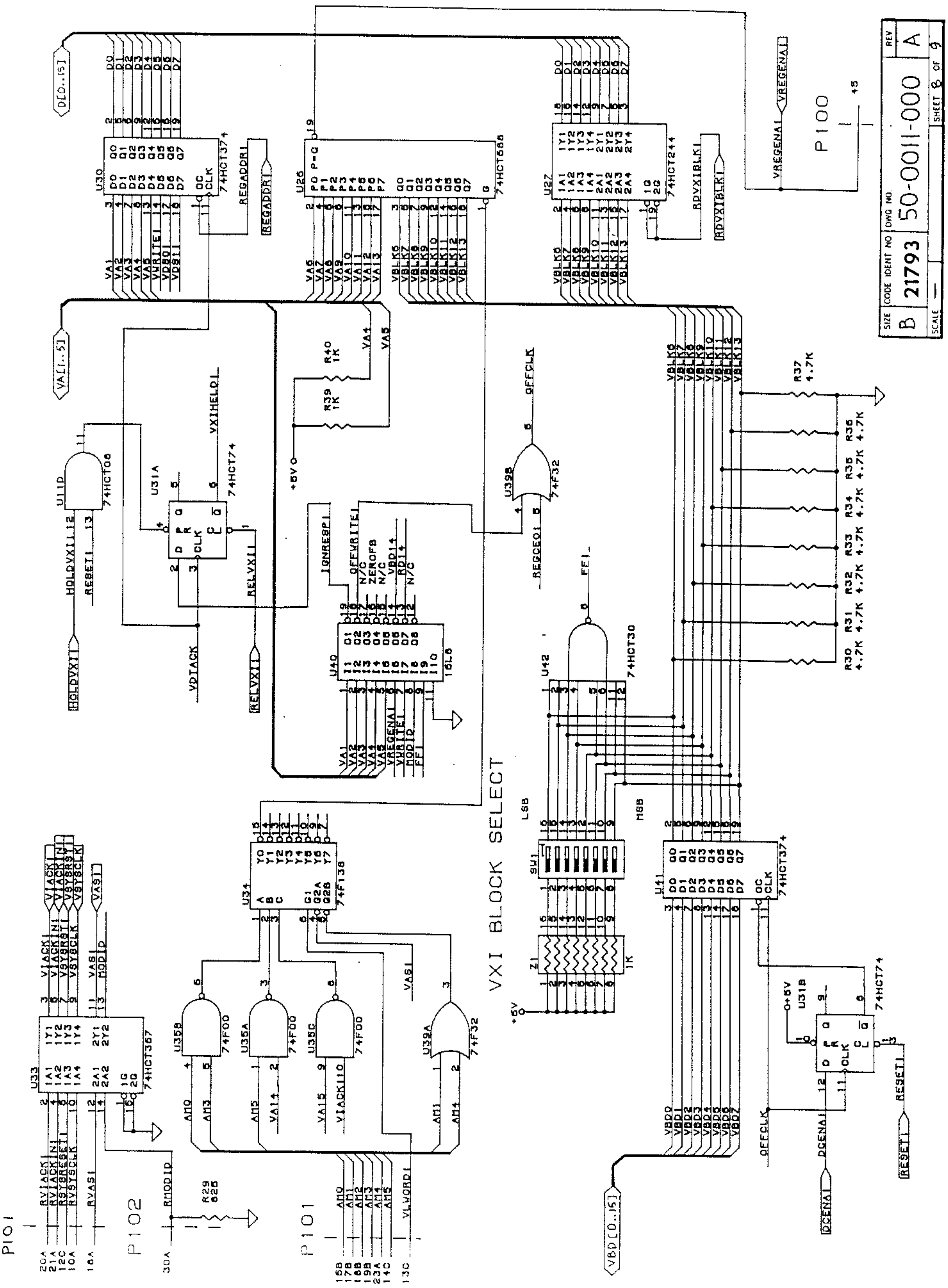
RESET
RESET1
R20
470
R22
4.7K
+5V
C1
0.01uF
U23
RESIN
CT
SENSE
RES
RES
REF
7705
+5V
C2
1.0uF
VSYSRST1
VSYSRST1
P101
A16
C10
VDTACK1
VSYSFAIL1
U8B
74S38
U8C
74S38
VDTACK
VDTACK
SYSFAIL
+5V
R23
330
R24
330
P1
HEADER
2X2
R25
100K
C3
1.0uF
Q1
MMBT3904
R27
1K
Q2
MMBT3904
R26
1K
R28
4.7K
FAILLED
U25
LM555
TR
R
Q
DIS
THR
CV
C4
0.01uF
VREGENA1
VREGENA1
IPL0
IPL1
IPL2
IPL01
IPL11
IPL21
U14
74HC148
A0
A1
A2
GS
EO
EI
R19
1K
R9
4.7K
R10
4.7K
R11
4.7K
R12
4.7K
R13
4.7K
R14
4.7K
R15
4.7K
R16
4.7K
CTIRQ1
IRQ21
LBIRQ1
IRQ41
IRQ1IRQ1
VXIHELD1
NMI1
IRQ21
LBIRQ1
IRQ41
VXIHELD1
SIZE
B
CODE IDENT NO
21793
DWG NO.
50-0011-000
REV
A
SCALE
SHEET 6 OF 9

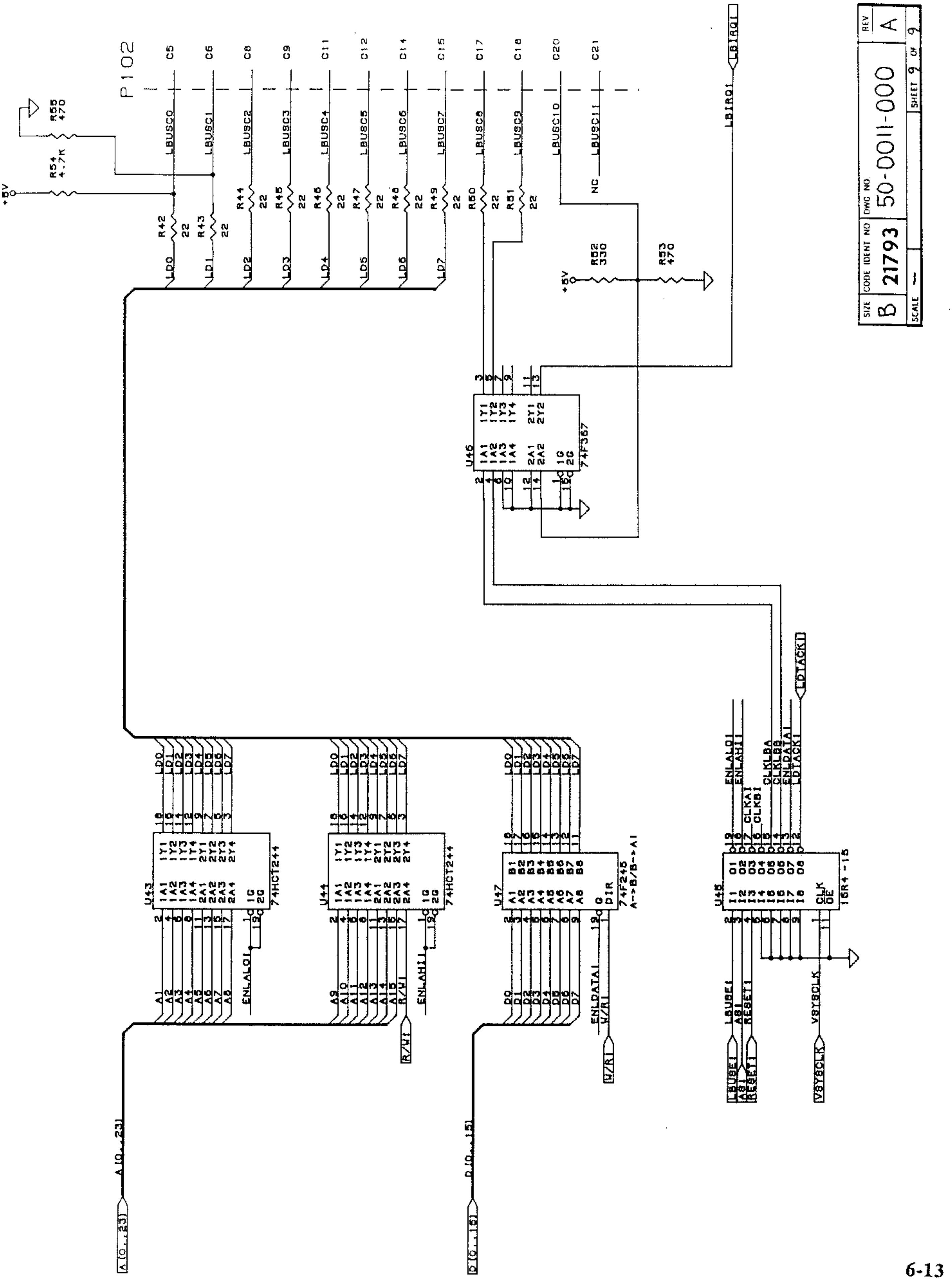


VXI REGISTER MAP

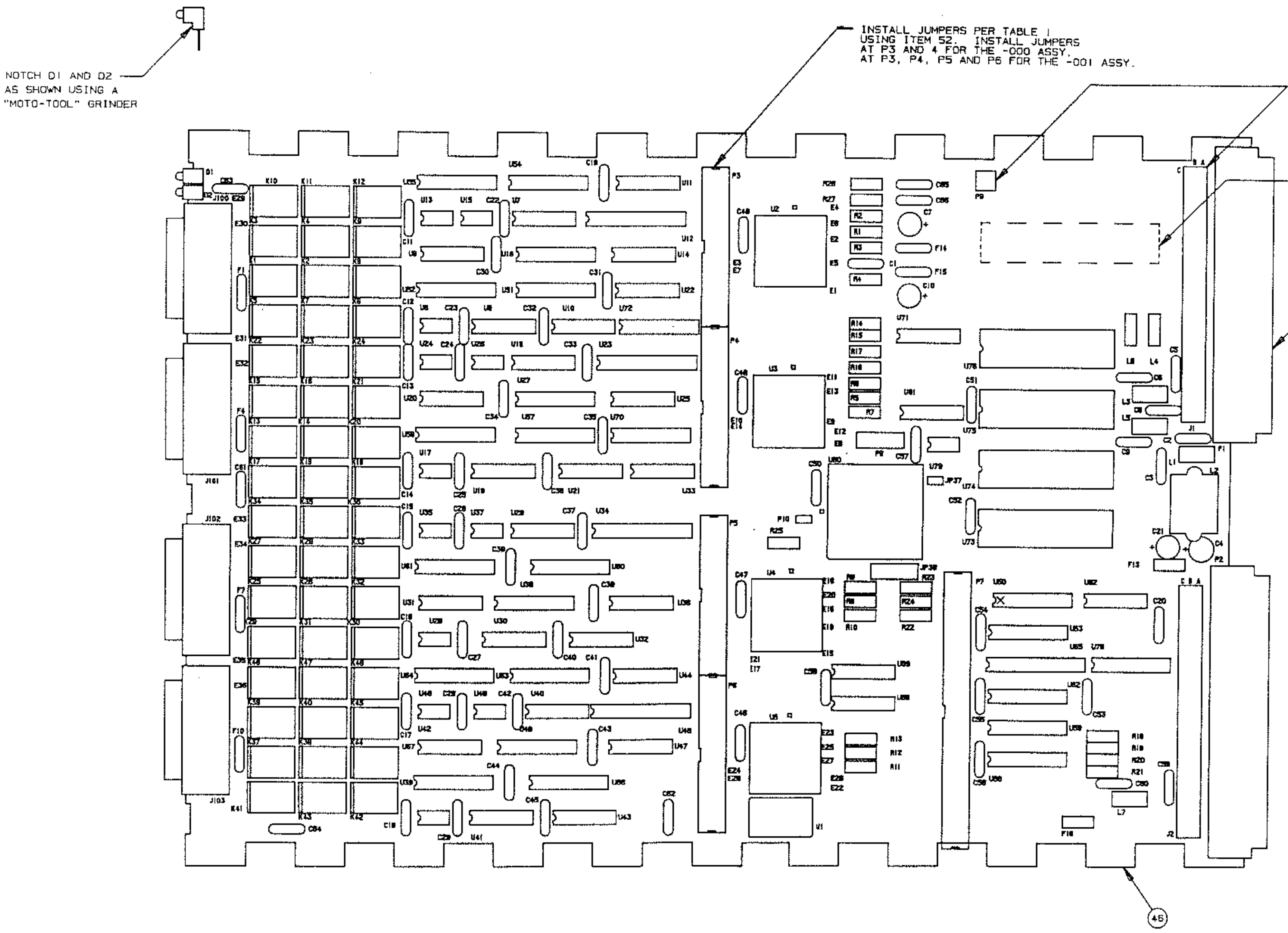
3FH	DEVICE DEPENDANT REGISTERS
1FH 18H	VXI RESERVED REGISTERS
14H	A32 POINTER
10H	A24 POINTER
0EH	DATA LOW
0CH	DATA HIGH
0AH	RESPONSE/DATA EXTENDED
08H	PROTOCOL/SIGNAL REGISTER
06H	OFFSET REGISTER
04H	STATUS/CONTROL REGISTER
02H	DEVICE TYPE
00H	ID/LOGICAL ADDRESS REGISTER

SIZE	CODE IDENT NO	DWG NO.	REV
B	21793	50-0011-000	A
SCALE —		SHEET 7 OF 9	


VXI BLOCK SELECT
SIZE B
CODE IDENT NO 21793
DWG NO. 50-0011-000
REV A
SCALE —
SHEET 8 OF 9
U30 74HCT374
U26 74HCT688
U27 74HCT244
U31A 74HCT74
U11D 74HCT08
U39B 74F32
U40 16L8
U42 74HCT30
U41 74HCT374
U31B 74HCT74
U33 74HCT367
U34 74F138
U35B 74F00
U35A 74F00
U35C 74F00
U39A 74F32
SW1
Z1 1K
R29 825
R30 4.7K
R31 4.7K
R32 4.7K
R33 4.7K
R34 4.7K
R35 4.7K
R36 4.7K
R37 4.7K
R39 1K
R40 1K
P100
P101
P102
LSB
MSB
+5V
D[0..15]
VA[1..5]
VBD[0..15]
REGADDR
RDVXIBLK
VREGENA
HOLDVXI
RELVXI
VDTACK
VXIHELD
IGNRESP
OFFWRITE
ZEROFB
VBD14
RD14
REGCEO
OFFCLK
FF
DCENA
RESET
RVIACK
RVIACKIN
RSYSRESET
RSYSCLK
RVAS
RMODID
VIACK
VIACKIN
VSYSRST
VSYSCLK
VAS
MODID
VLWORD
AM0
AM1
AM2
AM3
AM4
AM5


P102
C5
C6
C8
C9
C11
C12
C14
C15
C17
C18
C20
C21
LBUSC0
LBUSC1
LBUSC2
LBUSC3
LBUSC4
LBUSC5
LBUSC6
LBUSC7
LBUSC8
LBUSC9
LBUSC10
LBUSC11
NC
+5V
R54 4.7K
R55 470
R42 22
R43 22
R44 22
R45 22
R46 22
R47 22
R48 22
R49 22
R50 22
R51 22
R52 330
R53 470
LD0
LD1
LD2
LD3
LD4
LD5
LD6
LD7
LBIRQ1
U45
74F367
1A1
1A2
1A3
1A4
2A1
2A2
1G
2G
1Y1
1Y2
1Y3
1Y4
2Y1
2Y2
U43
74HCT244
U44
74HCT244
ENLALO1
ENLAHI1
A1
A2
A3
A4
A5
A6
A7
A8
A9
A10
A11
A12
A13
A14
A15
R/W1
U47
74F245
A->B/B->A1
G
DIR
B1
B2
B3
B4
B5
B6
B7
B8
D0
D1
D2
D3
D4
D5
D6
D7
ENLDATA1
W/R1
U46
16R4-15
CLK
OE
LBUSE1
AS1
RESET1
VSYSCLK
CLKA1
CLKB1
CLKLBA
CLKLBB
ENLDATA1
LDTACK1
A[0..23]
D[0..15]
SIZE
B
CODE IDENT NO
21793
DWG NO.
50-0011-000
REV
A
SCALE
SHEET 9 OF 9

DETAIL



NOTCH D1 AND D2 AS SHOWN USING A "MOTO-TOOL" GRINDER

INSTALL JUMPERS PER TABLE 1 USING ITEM 52. INSTALL JUMPERS AT P3 AND 4 FOR THE -000 ASSY. AT P3, P4, P5 AND P6 FOR THE -001 ASSY.

INSTALL P3A,B,C P4A,C AND P9 SQUARE TO THE PCB ASSURE THAT THE PINS ARE FULLY SEATED

SEE NOTE 1

INSTALL P1 AND P2 FLUSH AND SQUARE TO THE PCB.

TABLE 1

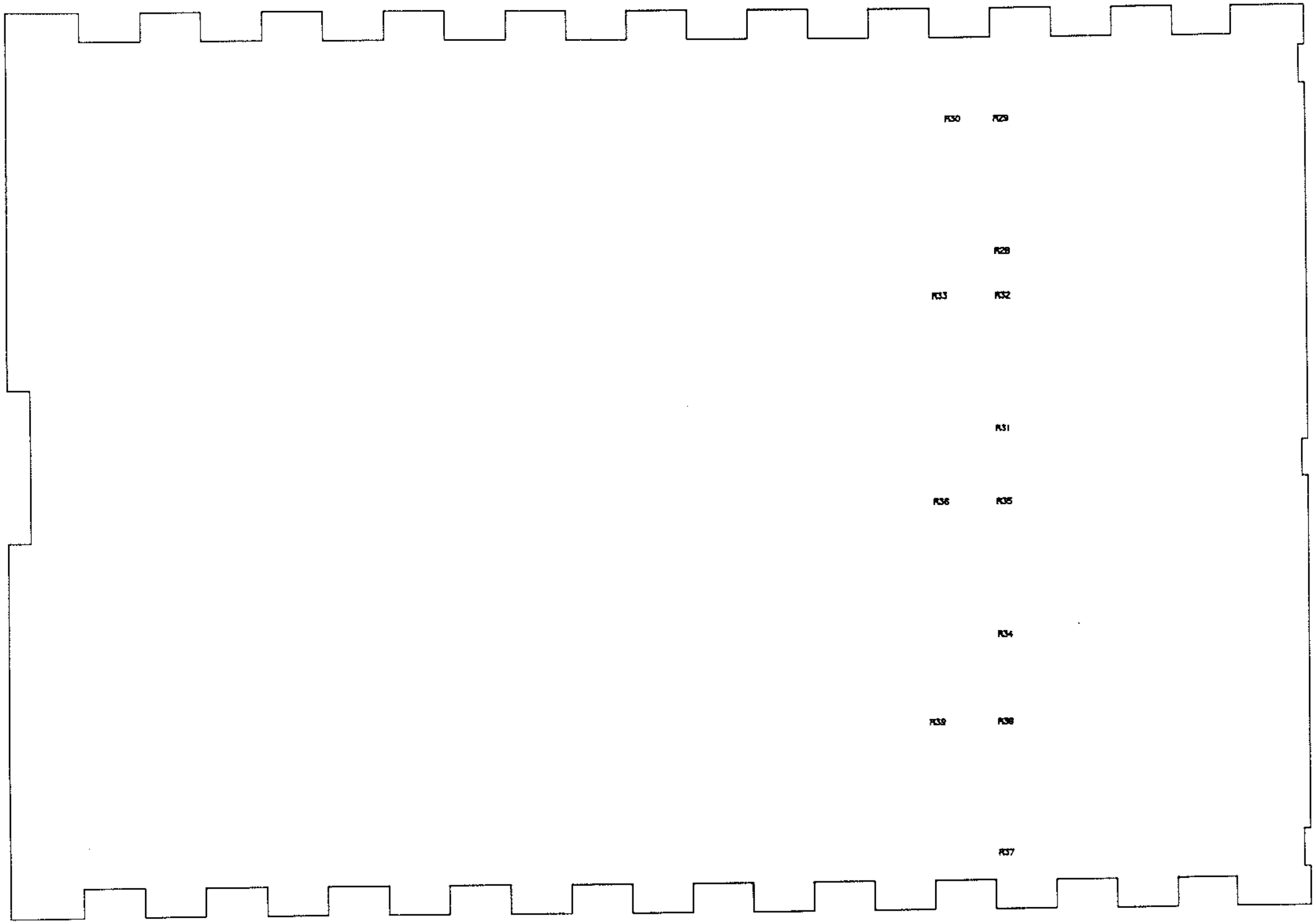
FROM	TO
1	2
3	4
5	6
7	8
9	10
11	12
13	14
15	16
17	18
19	20
21	22
23	24

3. FOR SCHEMATIC REFER TO 50-0031-000.
2. ASSEMBLE PER MIL-STD-454.
1. RUBBER STAMP ASSEMBLY NUMBER AND SERIAL NUMBER WITH INDELIBLE INK WHERE SHOWN 3/32" HIGH TEXT.

NOTES: UNLESS OTHERWISE SPECIFIED

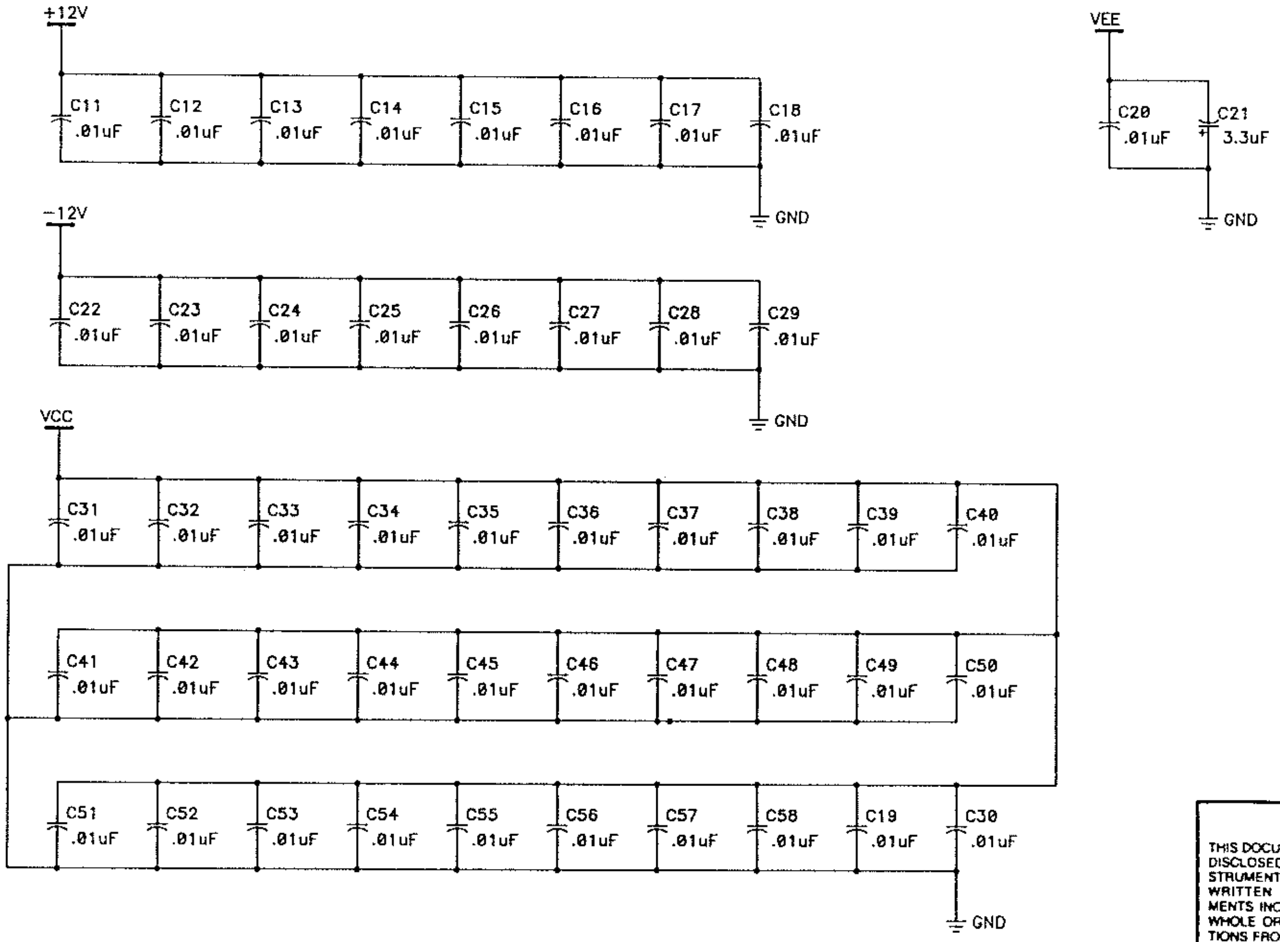
TITLE
ASSY, PCB, 4/8 CHANNEL VXI SERIAL INTERFACE

SIZE	CAGE CODE	DWG NO.	REV.
D		52-0033-000/001	A
SCALE NONE		SHEET 1 OF 6	



SOLDER SIDE OF PCB

SIZE	CAGE CODE	DWG NO.	REV.
D		52-0033-000/001	A
SCALE NONE		SHEET 2 OF 6	



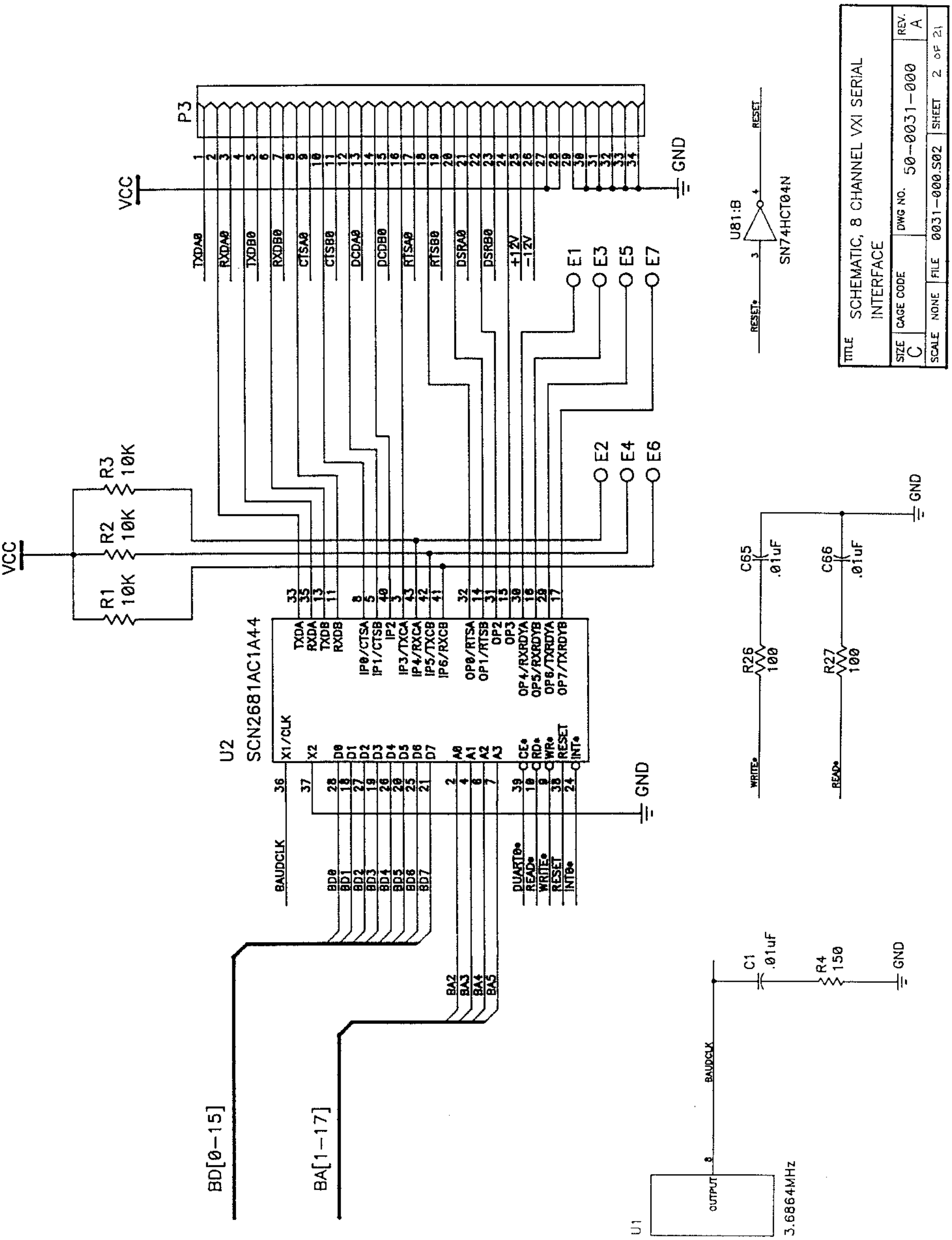
PROPRIETARY NOTICE

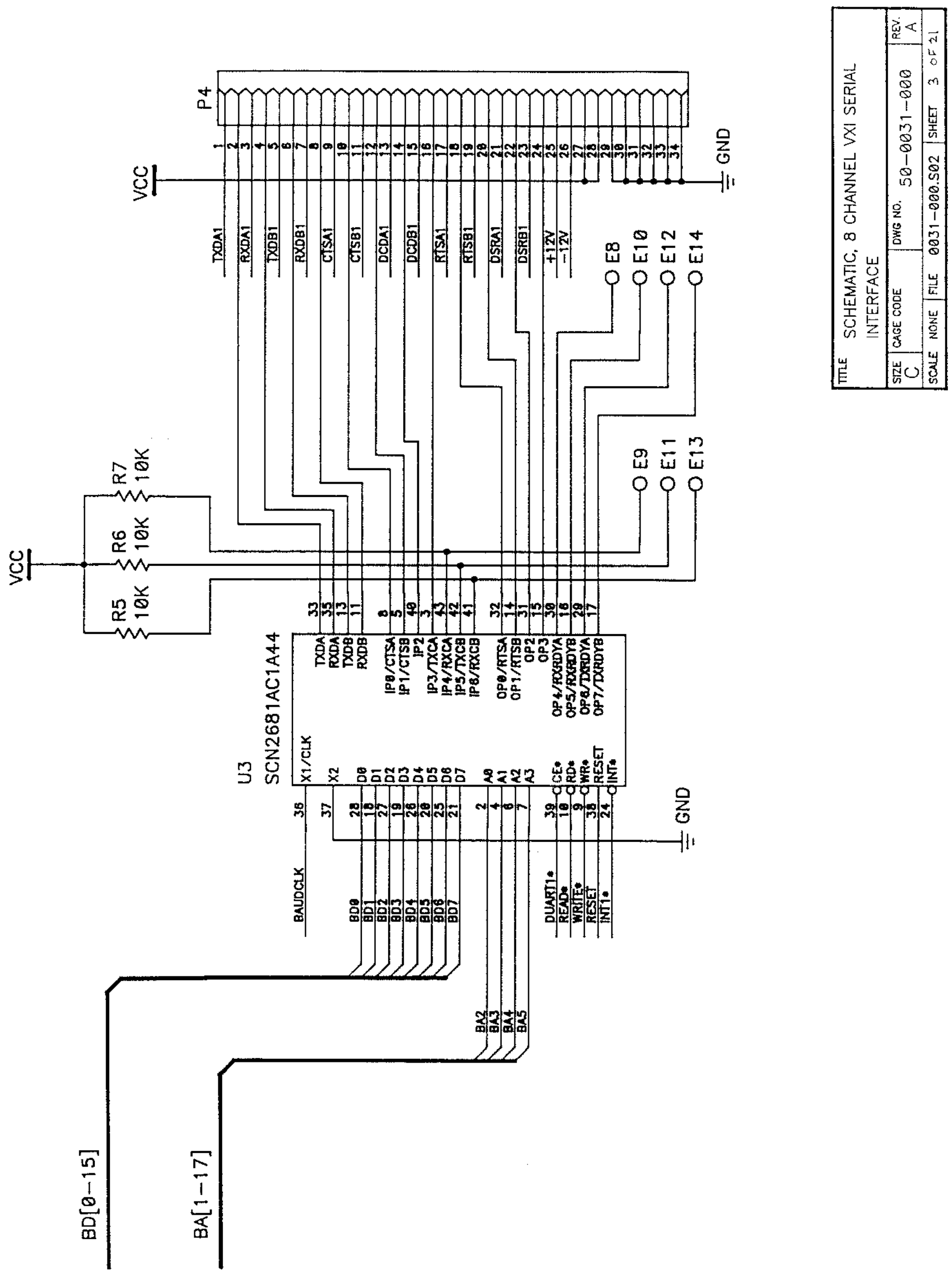
THIS DOCUMENT AND THE TECHNICAL DATA HEREON DISCLOSED, ARE PROPRIETARY TO RACAL-DANA INSTRUMENTS INC AND SHALL NOT, WITHOUT EXPRESS WRITTEN PERMISSION OF RACAL-DANA INSTRUMENTS INC. BE USED, RELEASED OR DISCLOSED IN WHOLE OR IN PART, OR USED TO SOLICIT QUOTATIONS FROM A COMPETITIVE SOURCE OR USED FOR MANUFACTURE BY ANYONE OTHER THAN RACAL-DANA INSTRUMENTS INC. THE INFORMATION HEREON HAS BEEN DEVELOPED AT PRIVATE EXPENSE, AND MAY ONLY BE USED FOR PURPOSES OF ENGINEERING EVALUATION AND FOR INCORPORATION INTO TECHNICAL SPECIFICATIONS AND OTHER DOCUMENTS WHICH SPECIFY PROCUREMENT OF PRODUCTS FROM RACAL-DANA INSTRUMENTS INC

5. REFER TO 52-0033-000/001 FOR THE PCB ASSEMBLY.
4. REFER TO 33-0030-000 FOR THE PCB.
3. ALL RESISTORS ARE 1/4W 5% CARBON COMPOSITION.
2. CAPACITOR VALUES LESS THAN 1.0uF ARE 50V MONOLYTHIC.
1. CAPACITOR VALUES GREATER THAN 1.0uF ARE 16V TANTALUM.

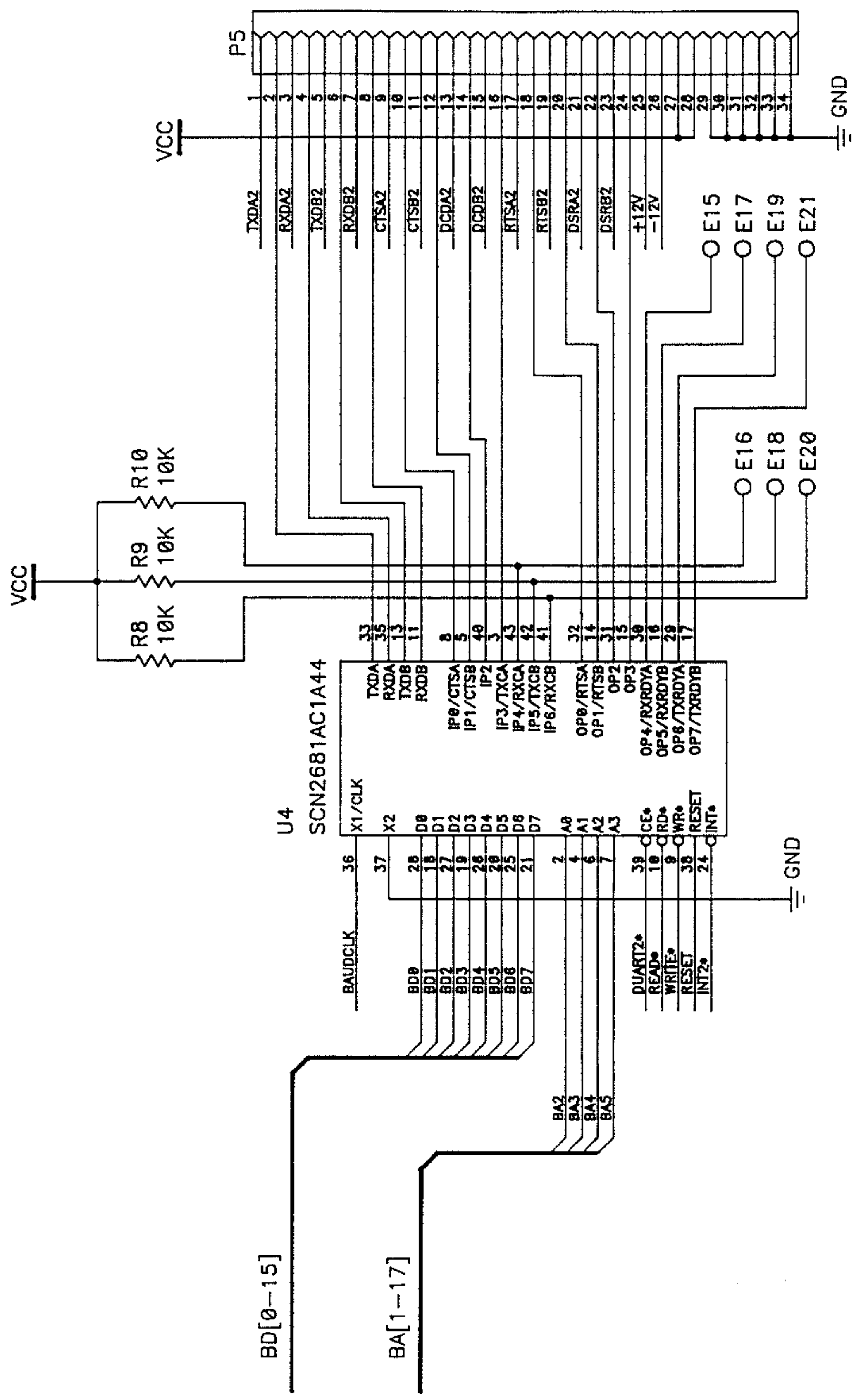
NOTES: UNLESS OTHER WISE SPECIFIED

TITLE SCHEMATIC, 8 CHANNEL VXI SERIAL INTERFACE			
SIZE C	CAGE CODE	DWG NO. 50-0031-000	REV. A
SCALE NONE	FILE 0031-000.S01	SHEET 1 OF 21	


TITLE SCHEMATIC, 8 CHANNEL VXI SERIAL INTERFACE
SIZE C
CAGE CODE
DWG NO. 50-0031-000
REV. A
SCALE NONE
FILE 0031-000.S02
SHEET 2 OF 21
P3
VCC
GND
TXDA0
RXDA0
TXDB0
RXDB0
CTSA0
CTSB0
DCDA0
DCDB0
RTSA0
RTSB0
DSRA0
DSRB0
+12V
-12V
E1
E3
E5
E7
E2
E4
E6
R1 10K
R2 10K
R3 10K
U2
SCN2681AC1A44
X1/CLK
X2
D0
D1
D2
D3
D4
D5
D6
D7
A0
A1
A2
A3
CE*
RD*
WR*
RESET
INT*
TXDA
RXDA
TXDB
RXDB
IP0/CTSA
IP1/CTSB
IP2
IP3/TXCA
IP4/RXCA
IP5/TXCB
IP6/RXCB
OP0/RTSA
OP1/RTSB
OP2
OP3
OP4/RXRDYA
OP5/RXRDYB
OP6/TXRDYA
OP7/TXRDYB
BAUDCLK
BD0
BD1
BD2
BD3
BD4
BD5
BD6
BD7
BA2
BA3
BA4
BA5
DUART0*
READ*
WRITE*
RESET
INT0*
BD[0-15]
BA[1-17]
U81:B
SN74HCT04N
RESET*
RESET
C65 .01uF
C66 .01uF
R26 100
R27 100
WRITE*
READ*
C1 .01uF
R4 150
U1
OUTPUT
3.6864MHz

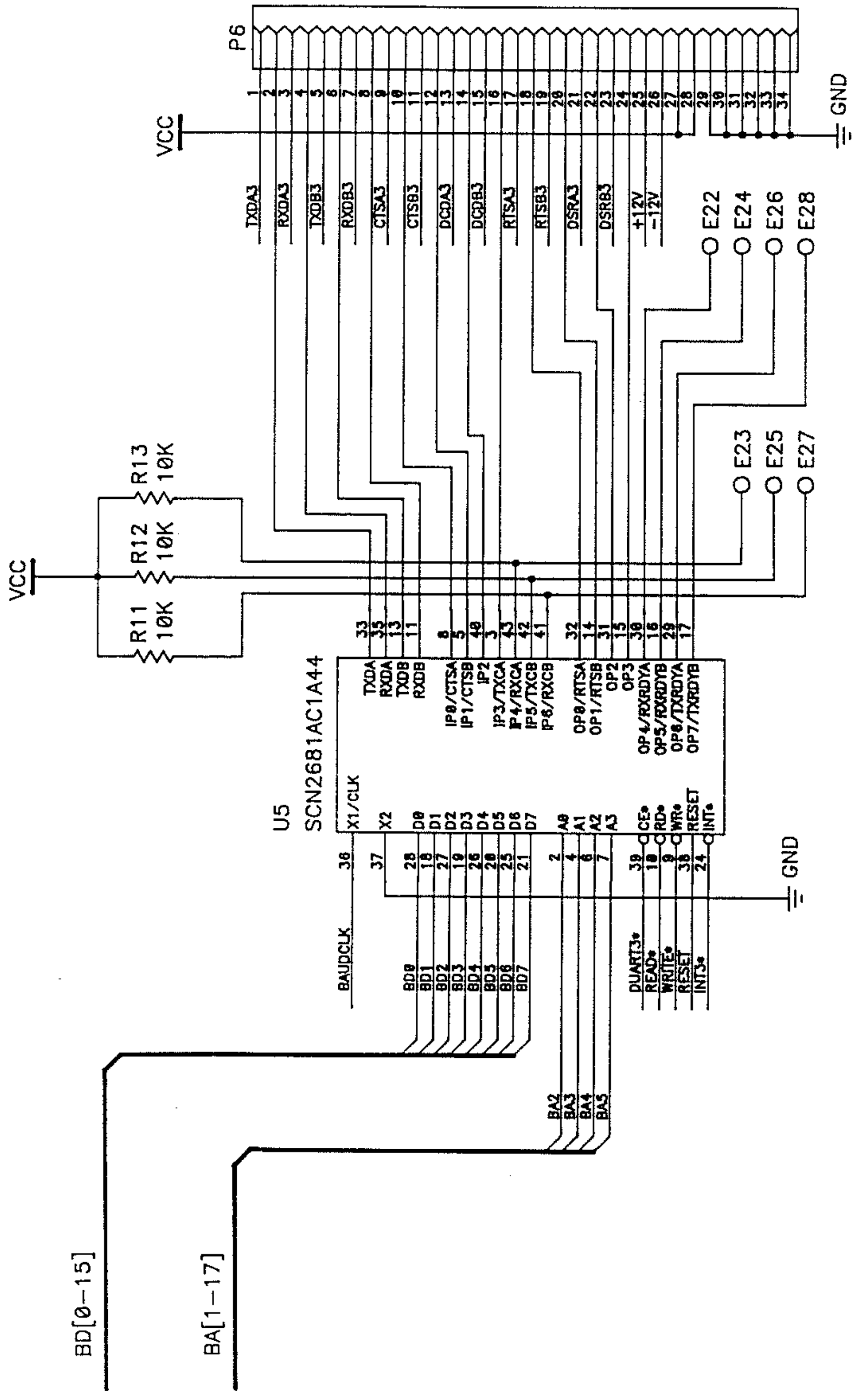


P4
VCC
GND
TXDA1
RXDA1
TXDB1
RXDB1
CTSA1
CTSB1
DCDA1
DCDB1
RTSA1
RTSB1
DSRA1
DSRB1
+12V
-12V
E8
E10
E12
E14
E9
E11
E13
R7 10K
R6 10K
R5 10K
VCC
U3
SCN2681AC1A44
X1/CLK
X2
D0 D1 D2 D3 D4 D5 D6 D7
A0 A1 A2 A3
CE* RD* WR* RESET INT*
TXDA RXDA TXDB RXDB
IP0/CTSA IP1/CTSB IP2
IP3/TXCA IP4/RXCA IP5/TXCB IP6/RXCB
OP0/RTSA OP1/RTSB OP2 OP3
OP4/RXRDYA OP5/RXRDYB OP6/TXRDYA OP7/TXRDYB
BAUDCLK
BD0 BD1 BD2 BD3 BD4 BD5 BD6 BD7
BA2 BA3 BA4 BA5
DUART1* READ* WRITE* RESET INT1*
BD[0-15]
BA[1-17]
GND
TITLE SCHEMATIC, 8 CHANNEL VXI SERIAL INTERFACE
SIZE C
CAGE CODE
DWG NO. 50-0031-000
REV. A
SCALE NONE
FILE 0031-000.S02
SHEET 3 OF 21

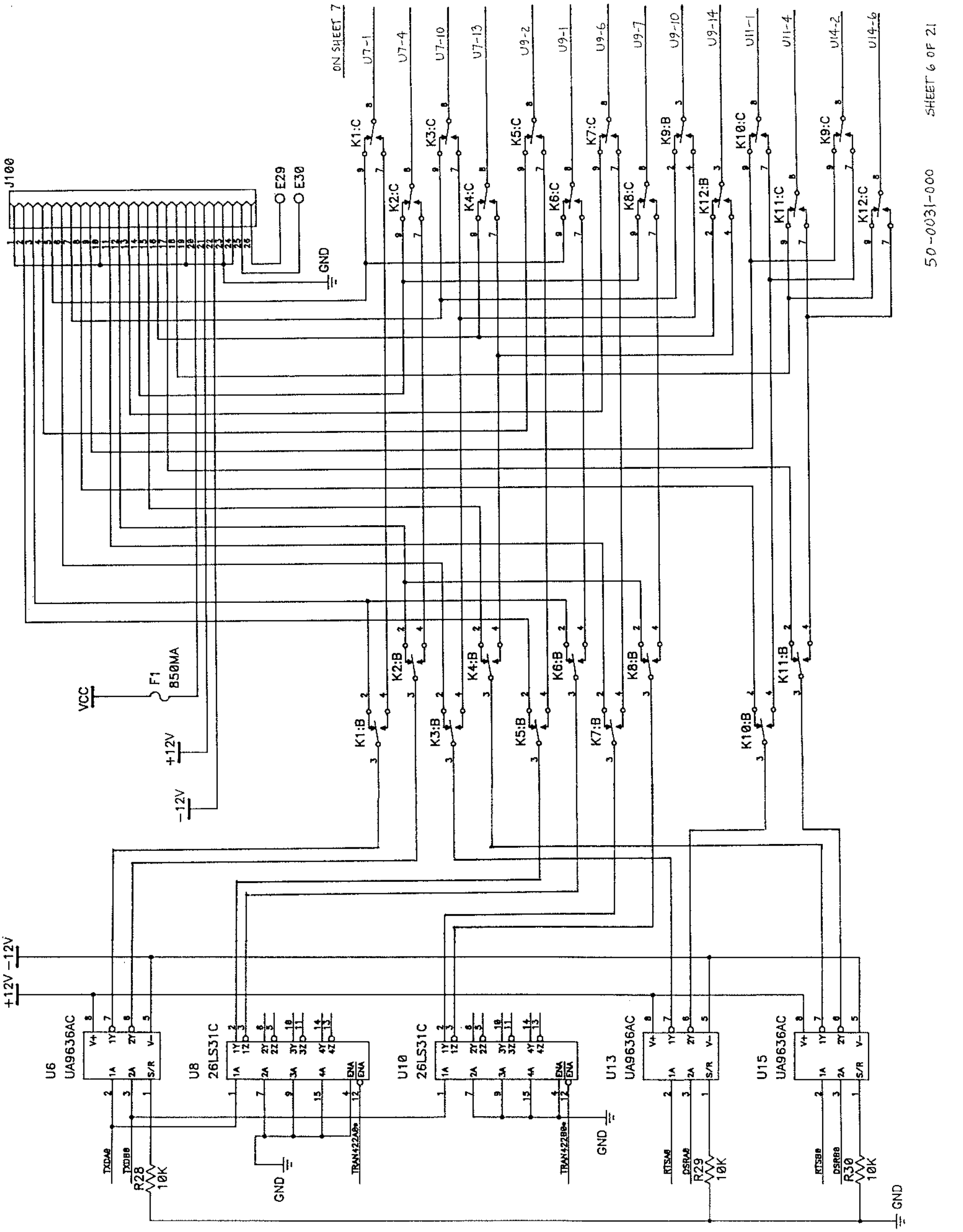

VCC
R8 10K
R9 10K
R10 10K
U4
SCN2681AC1A44
X1/CLK
X2
D0 D1 D2 D3 D4 D5 D6 D7
A0 A1 A2 A3
CE* RD* WR* RESET INT*
TXDA RXDA TXDB RXDB
IP0/CTSA IP1/CTSB IP2 IP3/TXCA IP4/RXCA IP5/TXCB IP6/RXCB
OP0/RTSA OP1/RTSB OP2 OP3 OP4/RXRDYA OP5/RXRDYB OP6/TXRDYA OP7/TXRDYB
BAUDCLK
BD0 BD1 BD2 BD3 BD4 BD5 BD6 BD7
BA2 BA3 BA4 BA5
DUART2* READ* WRITE* RESET INT2*
BD[0-15]
BA[1-17]
GND
E15 E16 E17 E18 E19 E20 E21
TXDA2 RXDA2 TXDB2 RXDB2 CTSA2 CTSB2 DCDA2 DCDB2 RTSA2 RTSB2 DSRA2 DSRB2 +12V -12V
P5
VCC
GND

TITLE SCHEMATIC, 8 CHANNEL VXI SERIAL INTERFACE

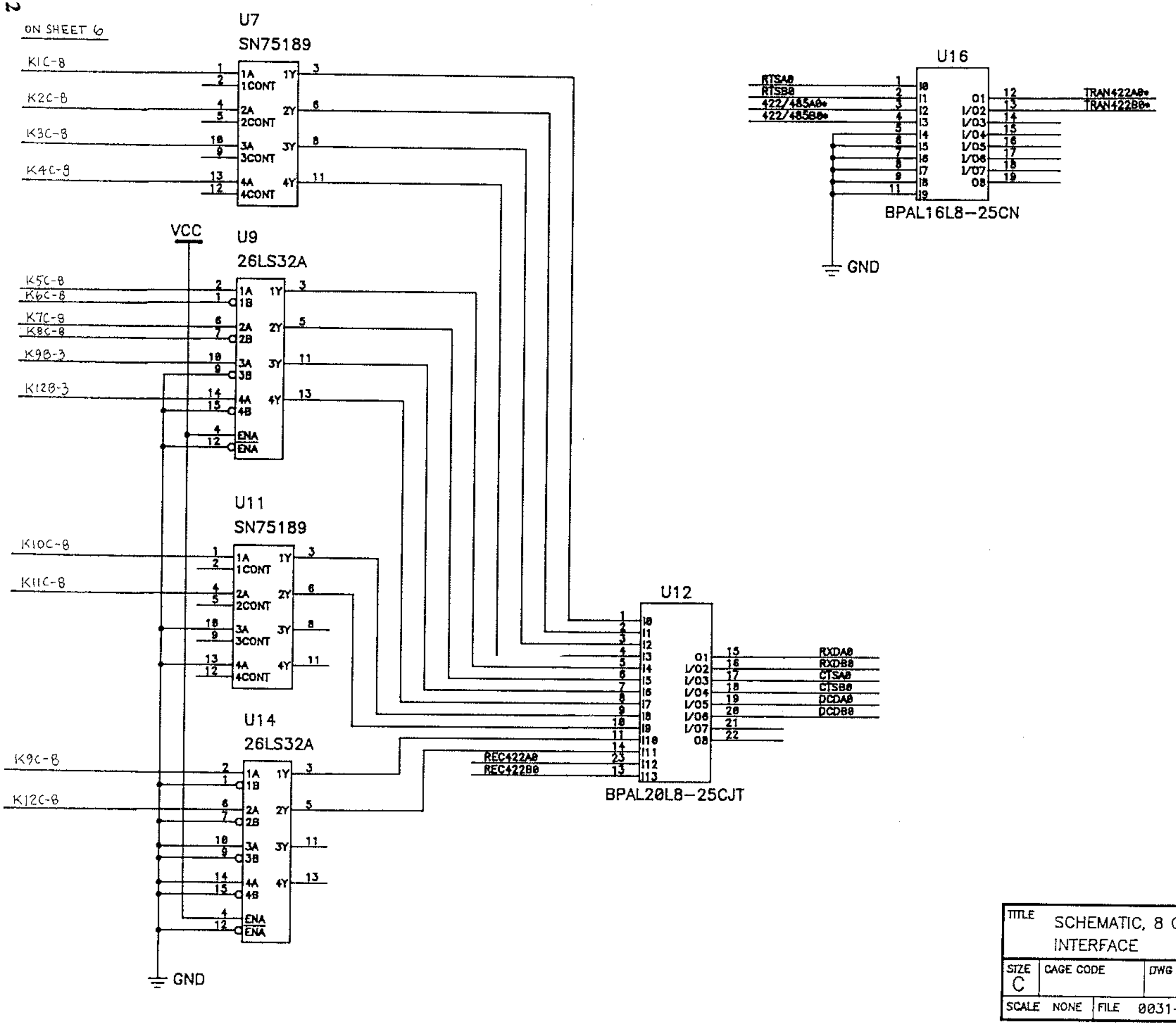
SIZE	CAGE CODE	DWG NO.	REV.
C		50-0031-000	A
SCALE NONE	FILE 0031-000.S02	SHEET 4 OF 21	



TITLE SCHEMATIC, 8 CHANNEL VXI SERIAL INTERFACE		
SIZE C	CAGE CODE	DWG NO. 50-0031-000	REV. A
SCALE NONE	FILE 0031-000.S02	SHEET 5 OF 21

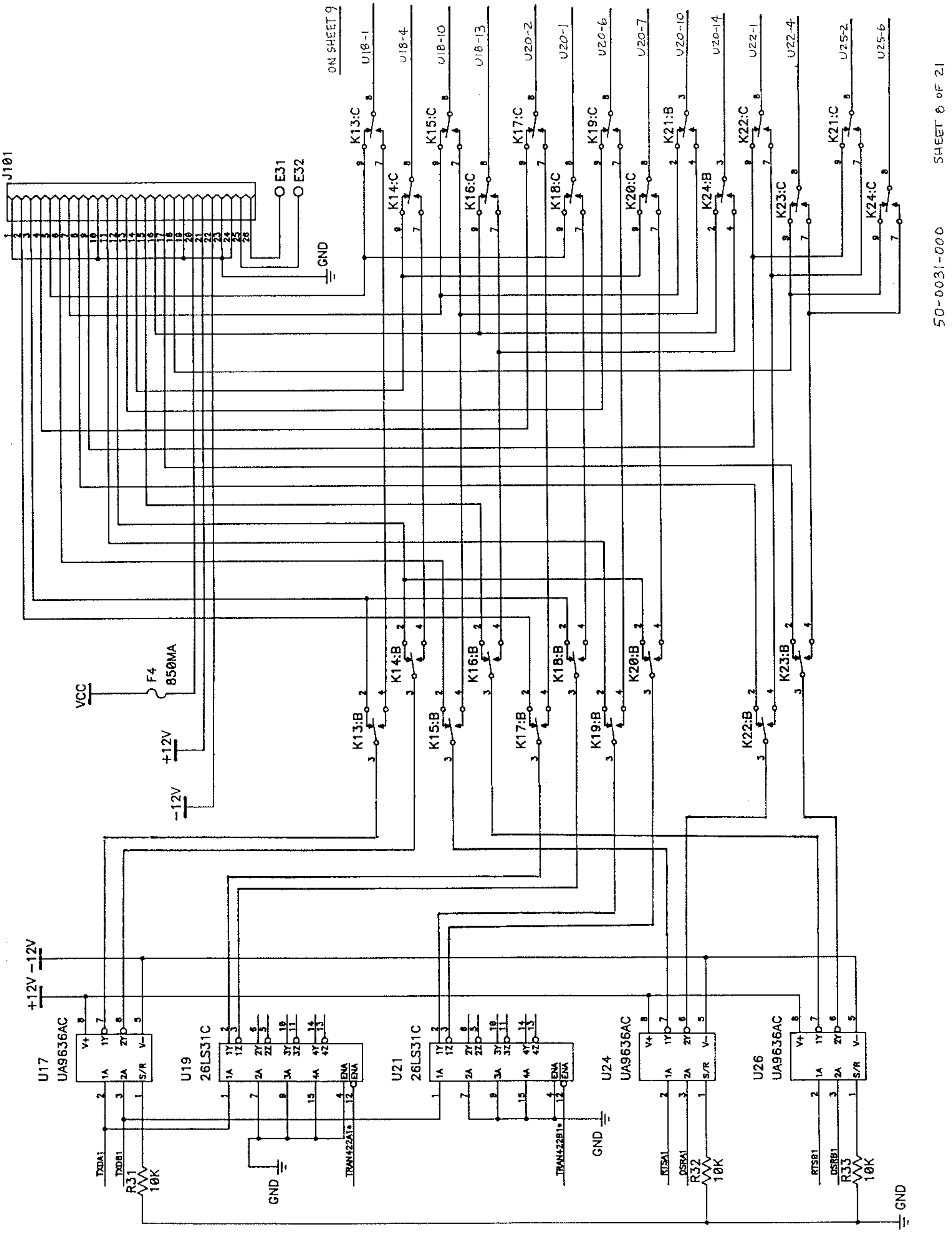


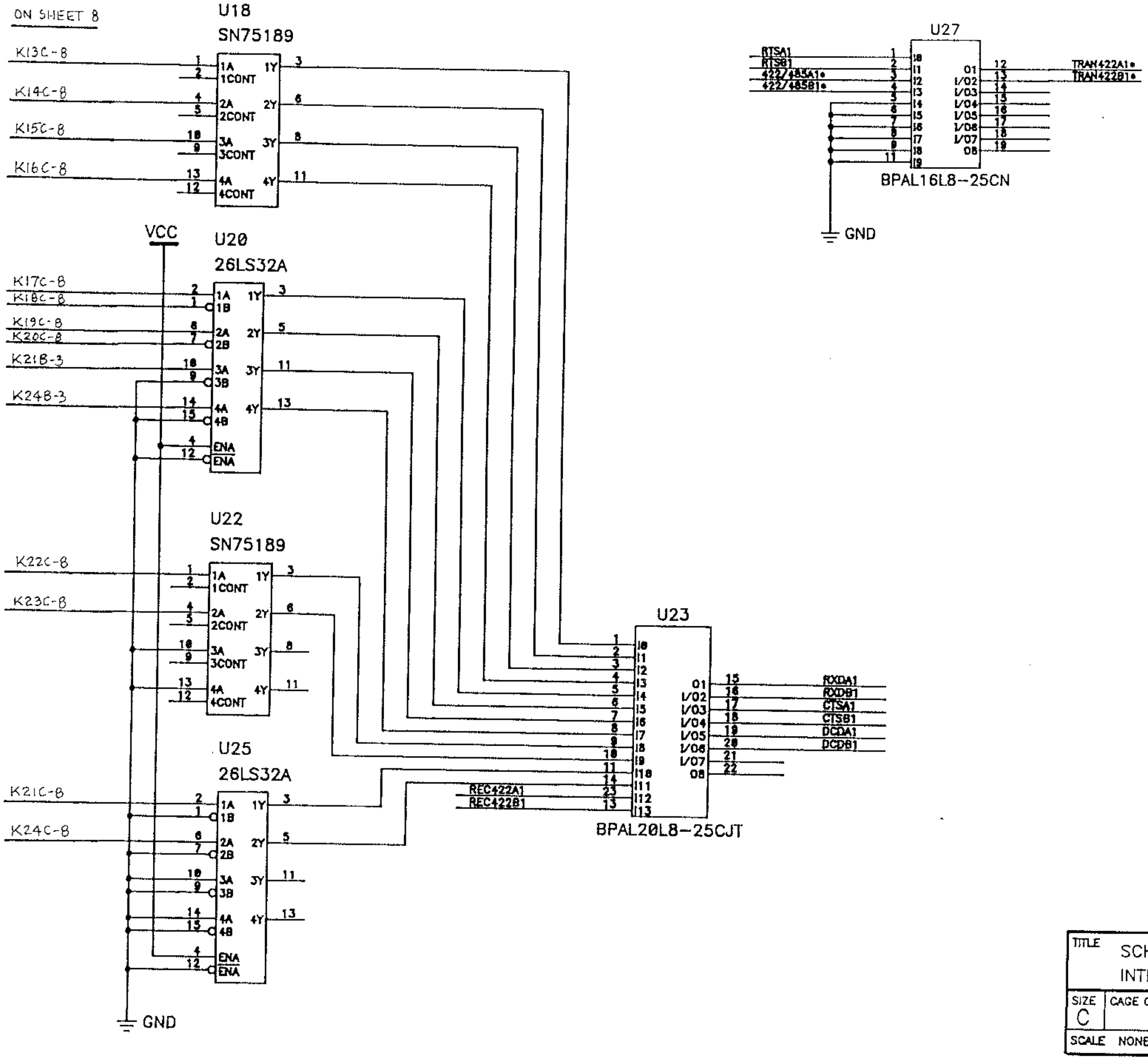
ON SHEET 7
U7-1
U7-4
U7-10
U7-13
U9-2
U9-1
U9-6
U9-7
U9-10
U9-14
U11-1
U11-4
U14-2
U14-6
SHEET 6 OF 21
50-0031-000
J100
E29
E30
GND
K1:C
K2:C
K3:C
K4:C
K5:C
K6:C
K7:C
K8:C
K9:B
K12:B
K10:C
K11:C
K9:C
K12:C
K1:B
K2:B
K3:B
K4:B
K5:B
K6:B
K7:B
K8:B
K10:B
K11:B
VCC
F1
850MA
+12V
-12V
+12V
-12V
U6
UA9636AC
U8
26LS31C
U10
26LS31C
U13
UA9636AC
U15
UA9636AC
TXDA0
TXDB0
R28
10K
GND
TRAN422A0*
TRAN422B0*
GND
RTSA0
DSRA0
R29
10K
RTSB0
DSRB0
R30
10K
GND



PIN	RS232	RS422
1	GND	GND
2	N/C	TXDA+
3	TXDA	TXDA−
4	N/C	RXDA+
5	RXDA	RXDA−
6	RTSA	RTSA
7	CTSA	CTSA
8	DSRA	DSRA
9	DCDA	DCDA
10	GND	GND
11	N/C	TXDB+
12	TXDB	TXDB−
13	N/C	RXDB+
14	RXDB	RXDB−
15	RTSB	RTSB
16	CTSB	CTSB
17	DSRB	DSRB
18	DCDB	DCDB
19	GND	GND
20	+5V	+5V
21	+12V	+12V
22	−12V	−12V
23	GND	GND
24	GND	GND
25	N/C	N/C
26	N/C	N/C

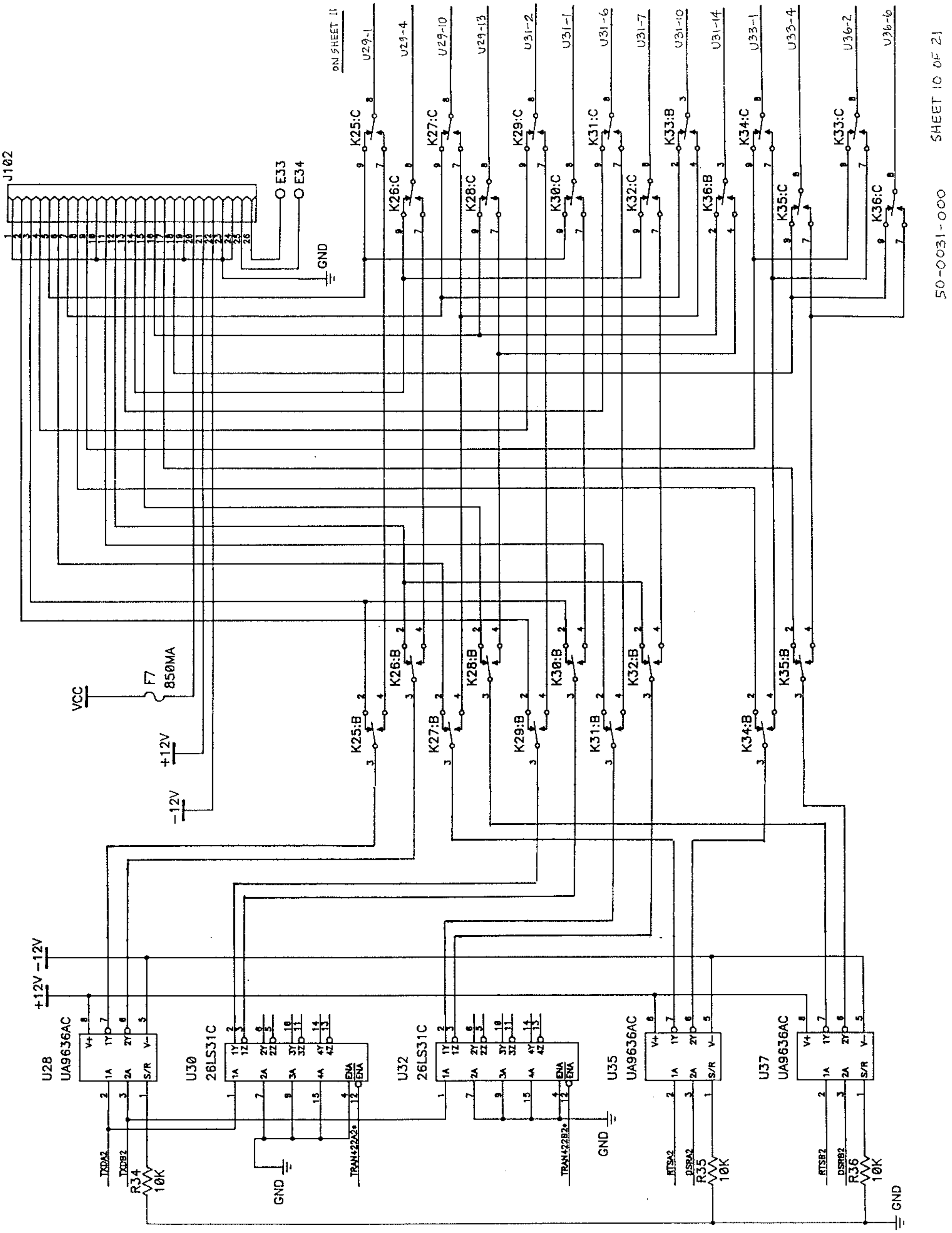
TITLE	SCHEMATIC, 8 CHANNEL VXI SERIAL INTERFACE		
SIZE C	CAGE CODE	DWG NO. 50−0031−000	REV. A
SCALE NONE	FILE 0031−000.S04		SHEET 7 OF 21

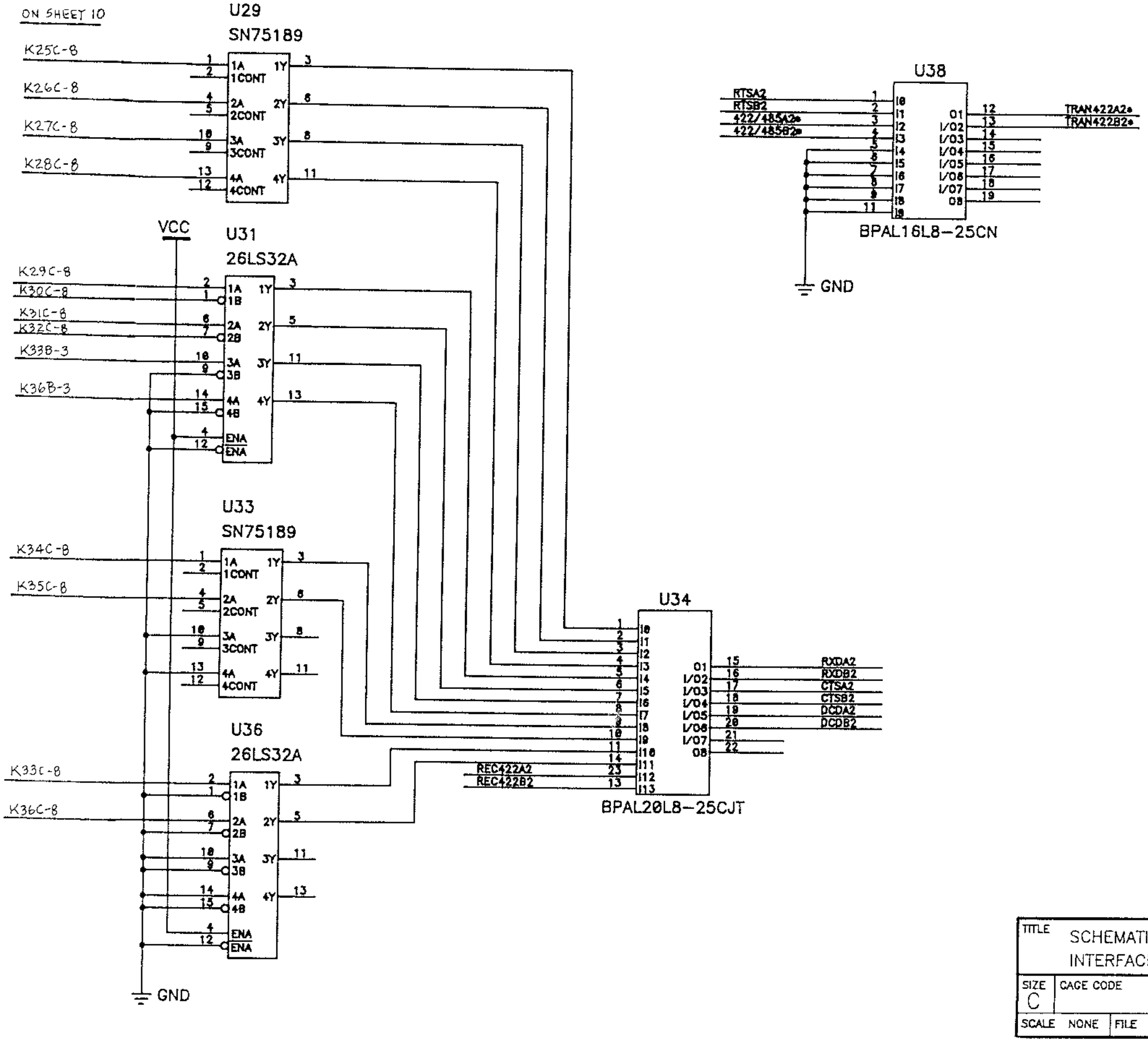

J101
E31
E32
GND
ON SHEET 9
U18-1
U18-4
U18-10
U18-13
U20-2
U20-1
U20-6
U20-7
U20-10
U20-14
U22-1
U22-4
U25-2
U25-6
K13:C
K14:C
K15:C
K16:C
K17:C
K18:C
K19:C
K20:C
K21:B
K24:B
K22:C
K23:C
K21:C
K24:C
SHEET 8 OF 21
50-0031-000
K13:B
K14:B
K15:B
K16:B
K17:B
K18:B
K19:B
K20:B
K22:B
K23:B
VCC
F4
850MA
+12V
-12V
U17
UA9636AC
U19
26LS31C
U21
26LS31C
U24
UA9636AC
U26
UA9636AC
TXDA1
TXDB1
R31
10K
TRAN422A1
TRAN422B1
RTSA1
DSRA1
R32
10K
RTSB1
DSRB1
R33
10K
GND



PIN	RS232	RS422
1	GND	GND
2	N/C	TXDA+
3	TXDA	TXDA−
4	N/C	RXDA+
5	RXDA	RXDA−
6	RTSA	RTSA
7	CTSA	CTSA
8	DSRA	DSRA
9	DCDA	DCDA
10	GND	GND
11	N/C	TXDB+
12	TXDB	TXDB−
13	N/C	RXDB+
14	RXDB	RXDB−
15	RTSB	RTSB
16	CTSB	CTSB
17	DSRB	DSRB
18	DCDB	DCDB
19	GND	GND
20	+5V	+5V
21	+12V	+12V
22	−12V	−12V
23	GND	GND
24	GND	GND
25	N/C	N/C
26	N/C	N/C

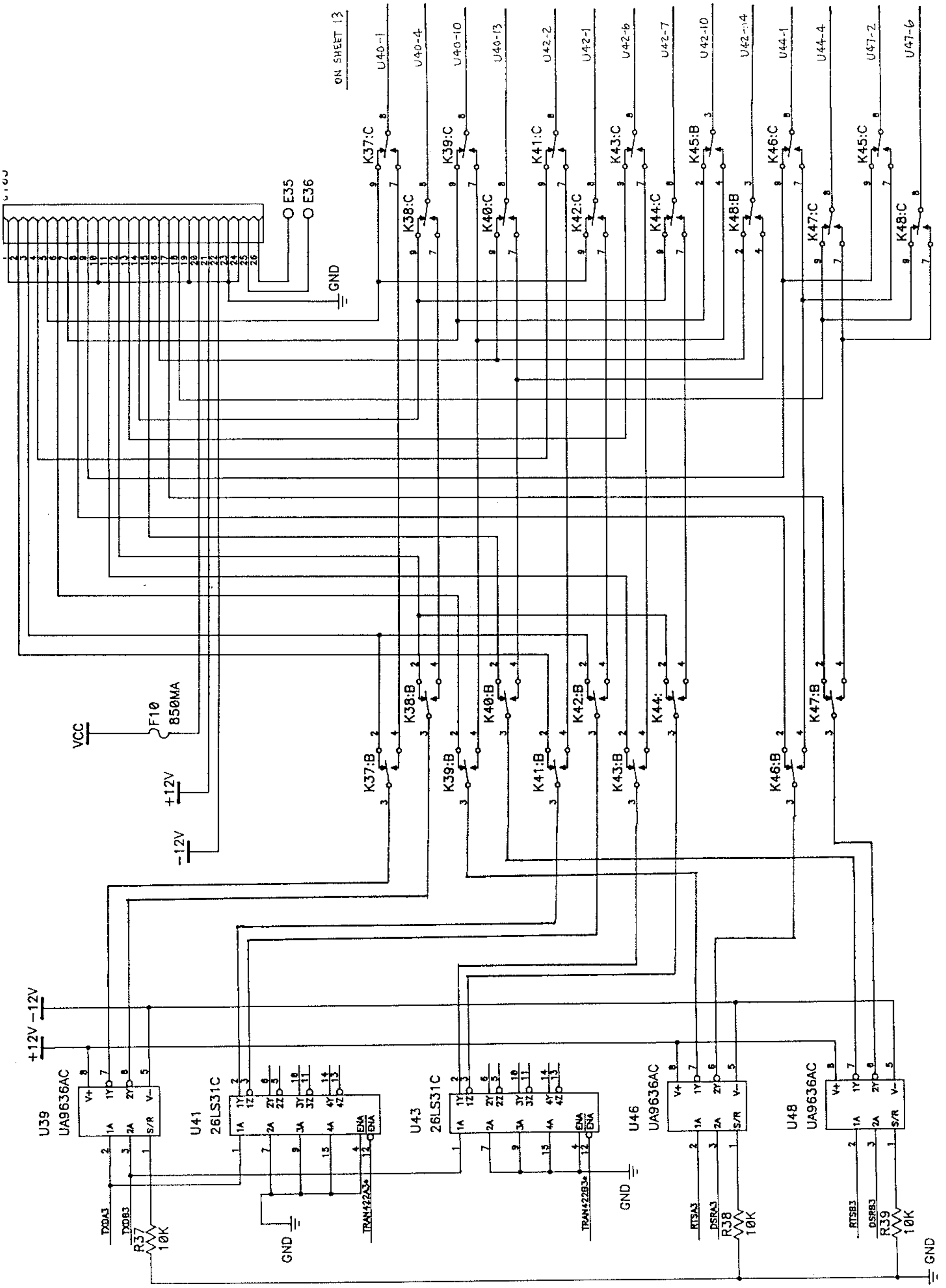
TITLE SCHEMATIC, 8 CHANNEL VXI SERIAL INTERFACE			
SIZE C	CAGE CODE	DWG NO. 50-0031-000	REV. A
SCALE NONE	FILE 0031-000.S05	SHEET 9 OF 21	


J102
E33
E34
GND
ON SHEET 11
U29-1
U29-4
U29-10
U29-13
U31-2
U31-1
U31-6
U31-7
U31-10
U31-14
U33-1
U33-4
U36-2
U36-6
SHEET 10 OF 21
50-0031-000
K25:C
K26:C
K27:C
K28:C
K29:C
K30:C
K31:C
K32:C
K33:B
K36:B
K34:C
K35:C
K33:C
K36:C
K25:B
K26:B
K27:B
K28:B
K29:B
K30:B
K31:B
K32:B
K34:B
K35:B
VCC
F7
850MA
+12V
-12V
U28
UA9636AC
U30
26LS31C
U32
26LS31C
U35
UA9636AC
U37
UA9636AC
TXDA2
TXDB2
R34
10K
TRAN422A2*
TRAN422B2*
RTSA2
DSRA2
R35
10K
RTSB2
DSRB2
R36
10K
GND



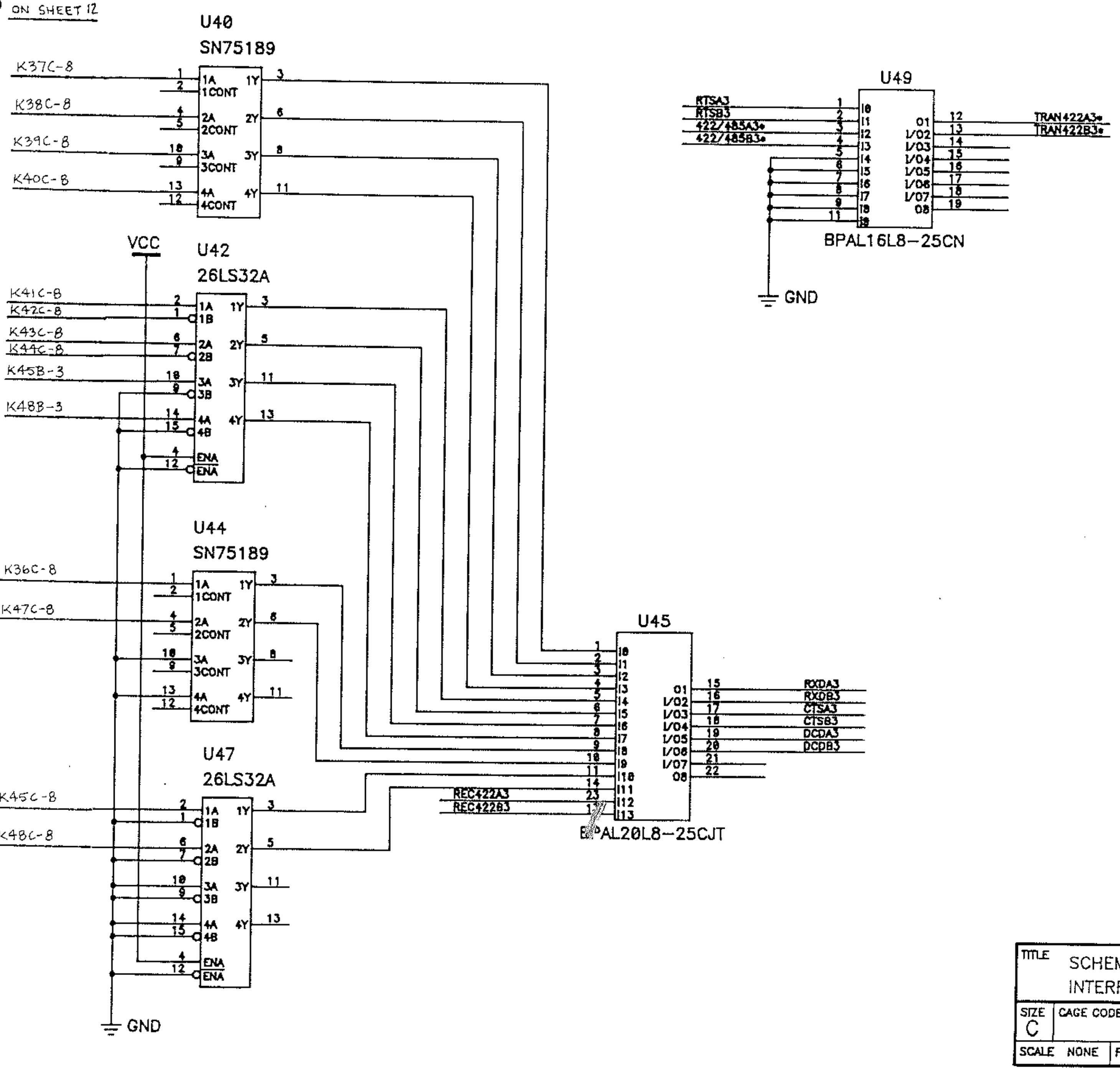
PIN	RS232	RS422
1	GND	GND
2	N/C	TXDA+
3	TXDA	TXDA−
4	N/C	RXDA+
5	RXDA	RXDA−
6	RTSA	RTSA
7	CTSA	CTSA
8	DSRA	DSRA
9	DCDA	DCDA
10	GND	GND
11	N/C	TXDB+
12	TXDB	TXDB−
13	N/C	RXDB+
14	RXDB	RXDB−
15	RTSB	RTSB
16	CTSB	CTSB
17	DSRB	DSRB
18	DCDB	DCDB
19	GND	GND
20	+5V	+5V
21	+12V	+12V
22	−12V	−12V
23	GND	GND
24	GND	GND
25	N/C	N/C
26	N/C	N/C

TITLE SCHEMATIC, 8 CHANNEL VXI SERIAL INTERFACE			
SIZE C	CAGE CODE	DWG NO. 50-0031-000	REV. A
SCALE NONE	FILE 0031-000.S06	SHEET 11 OF 21	



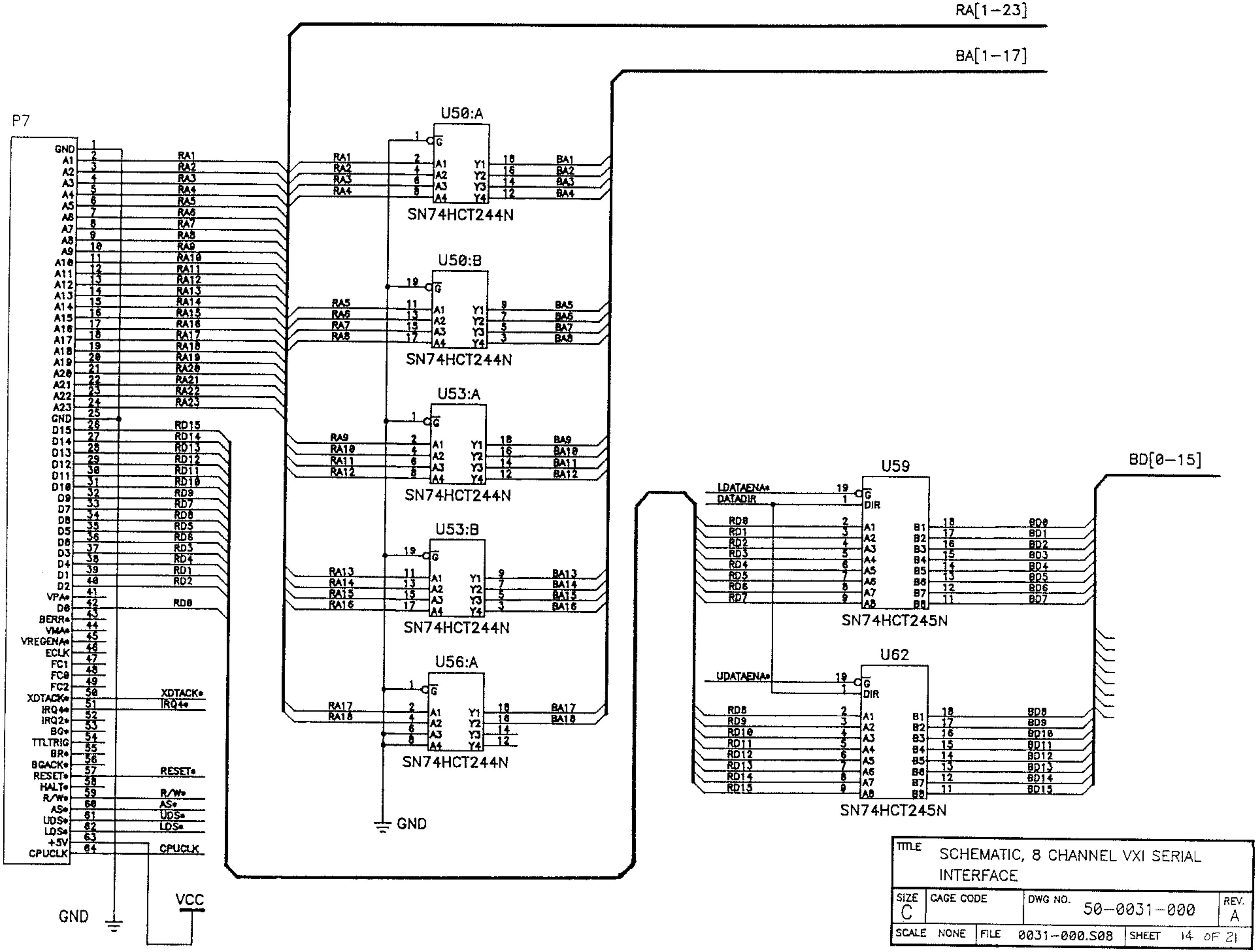
SHEET 12 OF 21

50-0031-000

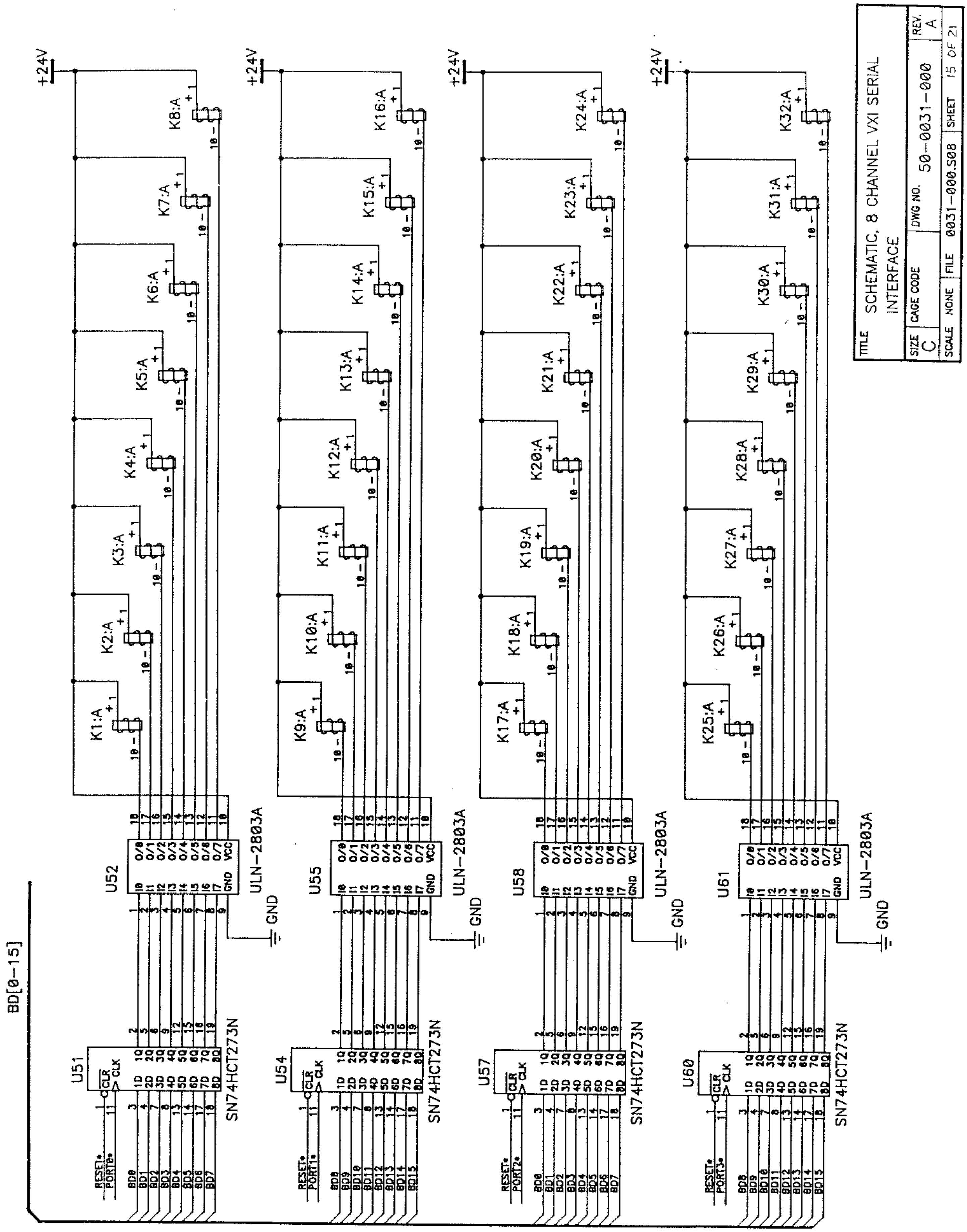


PIN	RS232	RS422
1	GND	GND
2	N/C	TXDA+
3	TXDA	TXDA–
4	N/C	RXDA+
5	RXDA	RXDA–
6	RTSA	RTSA
7	CTSA	CTSA
8	DSRA	DSRA
9	DCDA	DCDA
10	GND	GND
11	N/C	TXDB+
12	TXDB	TXDB–
13	N/C	RXDB+
14	RXDB	RXDB–
15	RTSB	RTSB
16	CTSB	CTSB
17	DSRB	DSRB
18	DCDB	DCDB
19	GND	GND
20	+5V	+5V
21	+12V	+12V
22	–12V	–12V
23	GND	GND
24	GND	GND
25	N/C	N/C
26	N/C	N/C

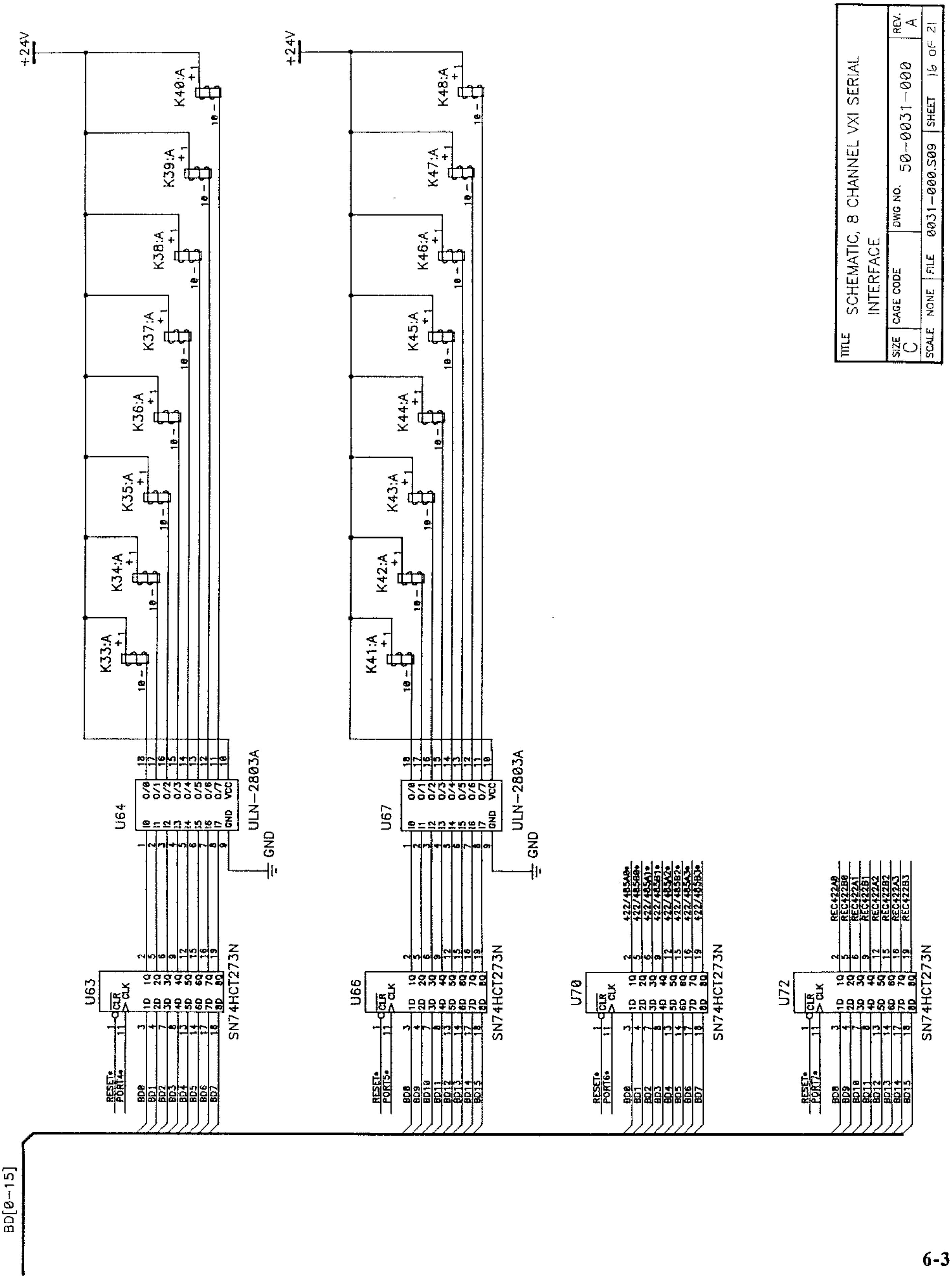
TITLE	SCHEMATIC, 8 CHANNEL VXI SERIAL INTERFACE		
SIZE C	CAGE CODE	DWG NO. 50–0031–000	REV. A
SCALE NONE	FILE 0031–000.S07	SHEET 13 OF 21	



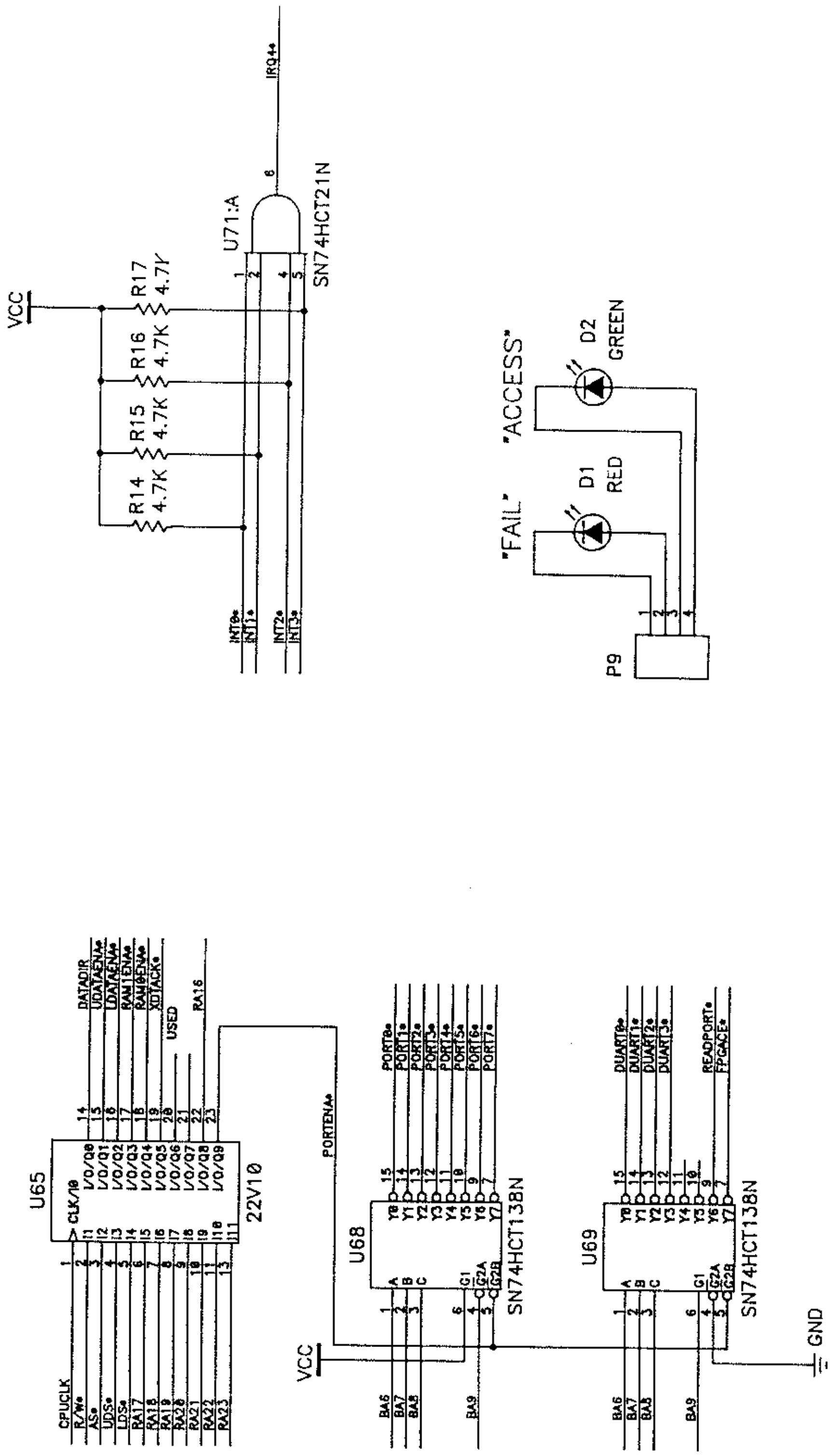
TITLE	SCHEMATIC, 8 CHANNEL VXI SERIAL INTERFACE		
SIZE C	CAGE CODE	DWG NO. 50-0031-000	REV. A
SCALE NONE	FILE 0031-000.S08	SHEET 14 OF 21	



TITLE SCHEMATIC, 8 CHANNEL VXI SERIAL INTERFACE
SIZE C
CAGE CODE
DWG NO. 50-0031-000
REV. A
SCALE NONE
FILE 0031-000.S08
SHEET 15 OF 21
+24V
K1:A
K2:A
K3:A
K4:A
K5:A
K6:A
K7:A
K8:A
K9:A
K10:A
K11:A
K12:A
K13:A
K14:A
K15:A
K16:A
K17:A
K18:A
K19:A
K20:A
K21:A
K22:A
K23:A
K24:A
K25:A
K26:A
K27:A
K28:A
K29:A
K30:A
K31:A
K32:A
U51
U52
U54
U55
U57
U58
U60
U61
ULN-2803A
SN74HCT273N
GND
VCC
CLR
CLK
BD[0-15]
RESET*
PORT0*
PORT1*
PORT2*
PORT3*
BD0
BD1
BD2
BD3
BD4
BD5
BD6
BD7
BD8
BD9
BD10
BD11
BD12
BD13
BD14
BD15
1D 2D 3D 4D 5D 6D 7D 8D
1Q 2Q 3Q 4Q 5Q 6Q 7Q 8Q
I0 I1 I2 I3 I4 I5 I6 I7
O/0 O/1 O/2 O/3 O/4 O/5 O/6 O/7

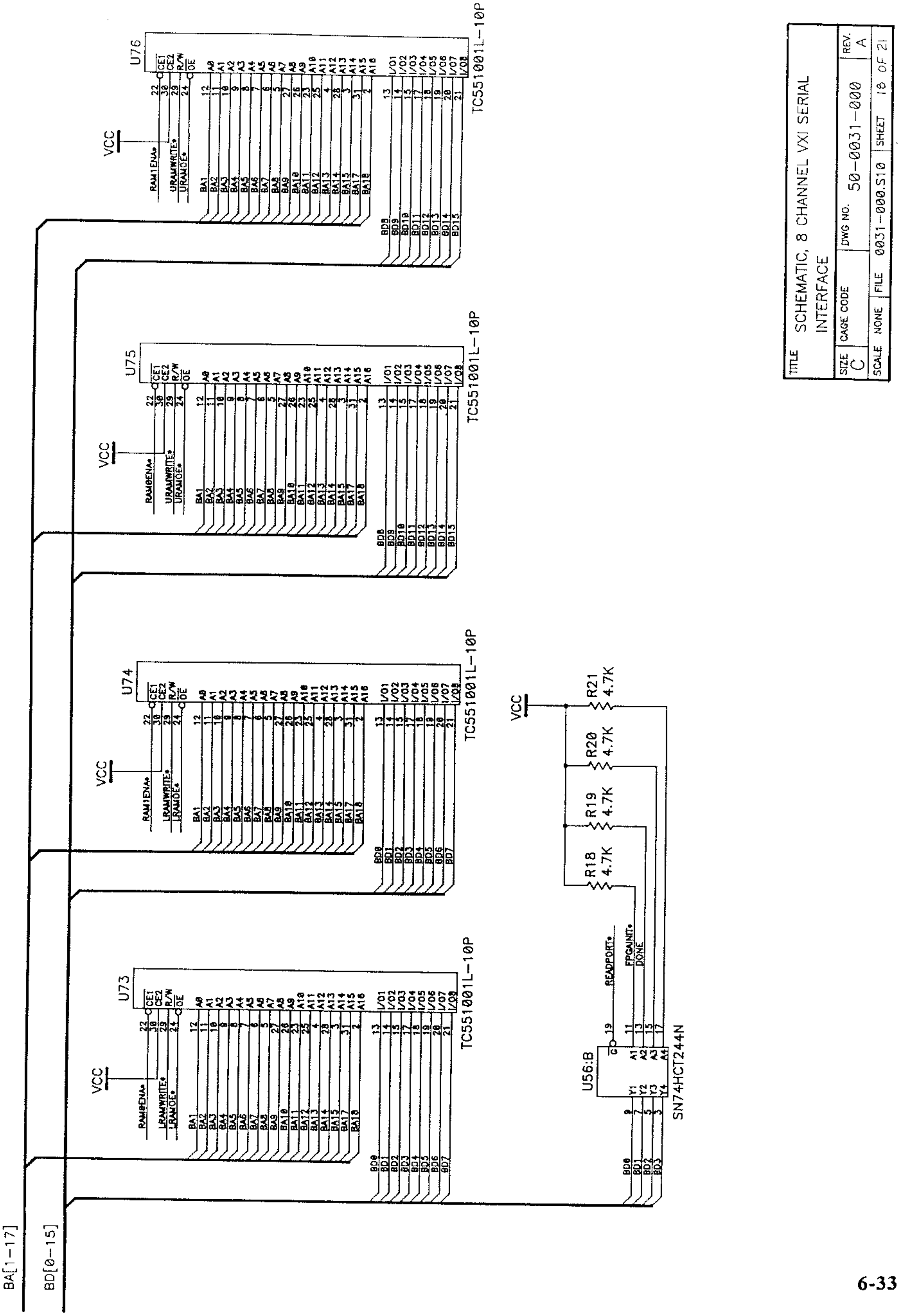


TITLE	SCHEMATIC, 8 CHANNEL VXI SERIAL INTERFACE		
SIZE C	CAGE CODE	DWG NO. 50-0031-000	REV. A
SCALE NONE	FILE 0031-000.S09		SHEET 16 OF 21



TITLE SCHEMATIC, 8 CHANNEL VXI SERIAL INTERFACE

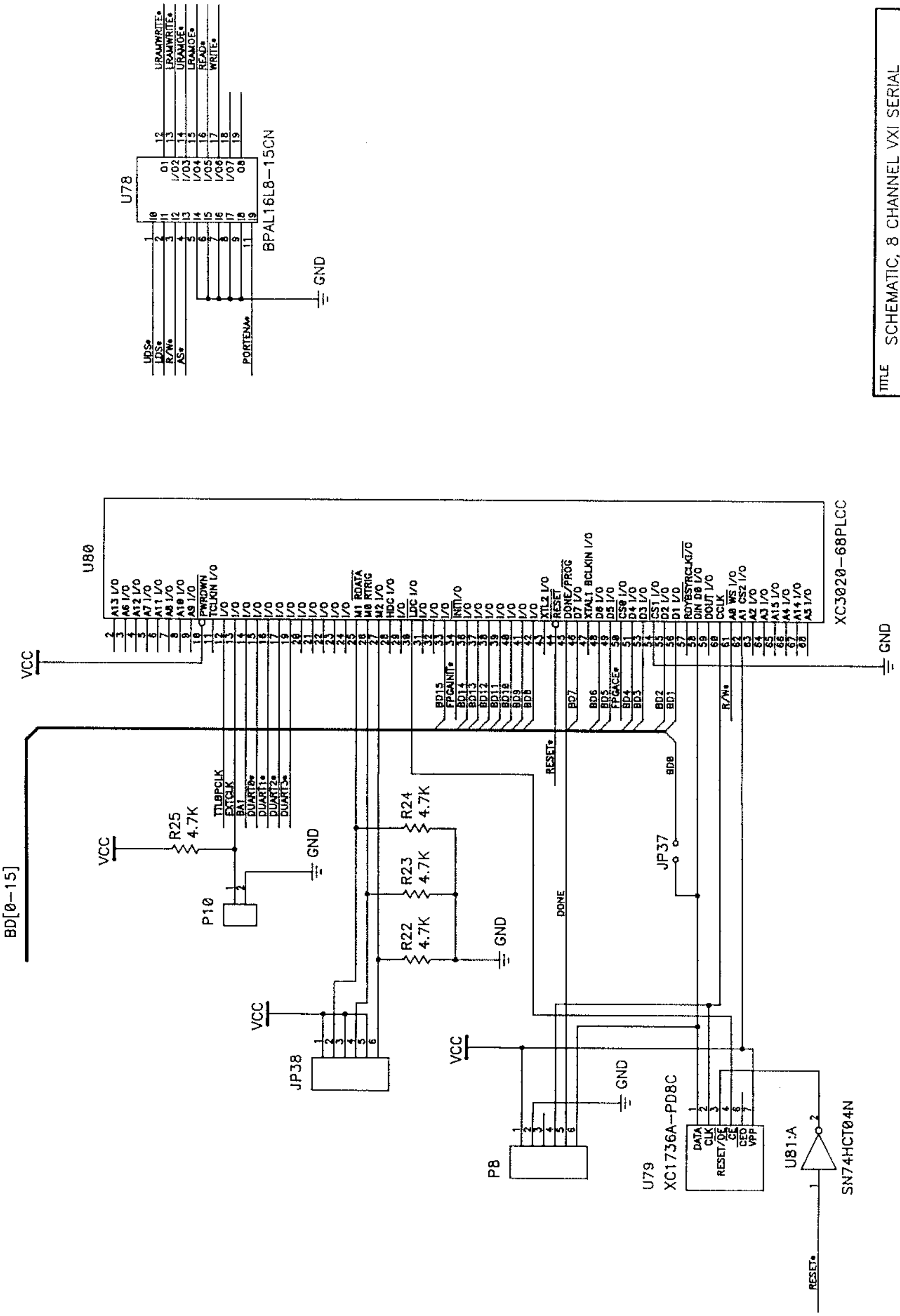
SIZE	CAGE CODE	DWG NO.	REV.
C		50-0031-000	A
SCALE NONE	FILE 0031-000.S09		SHEET 17 OF 21



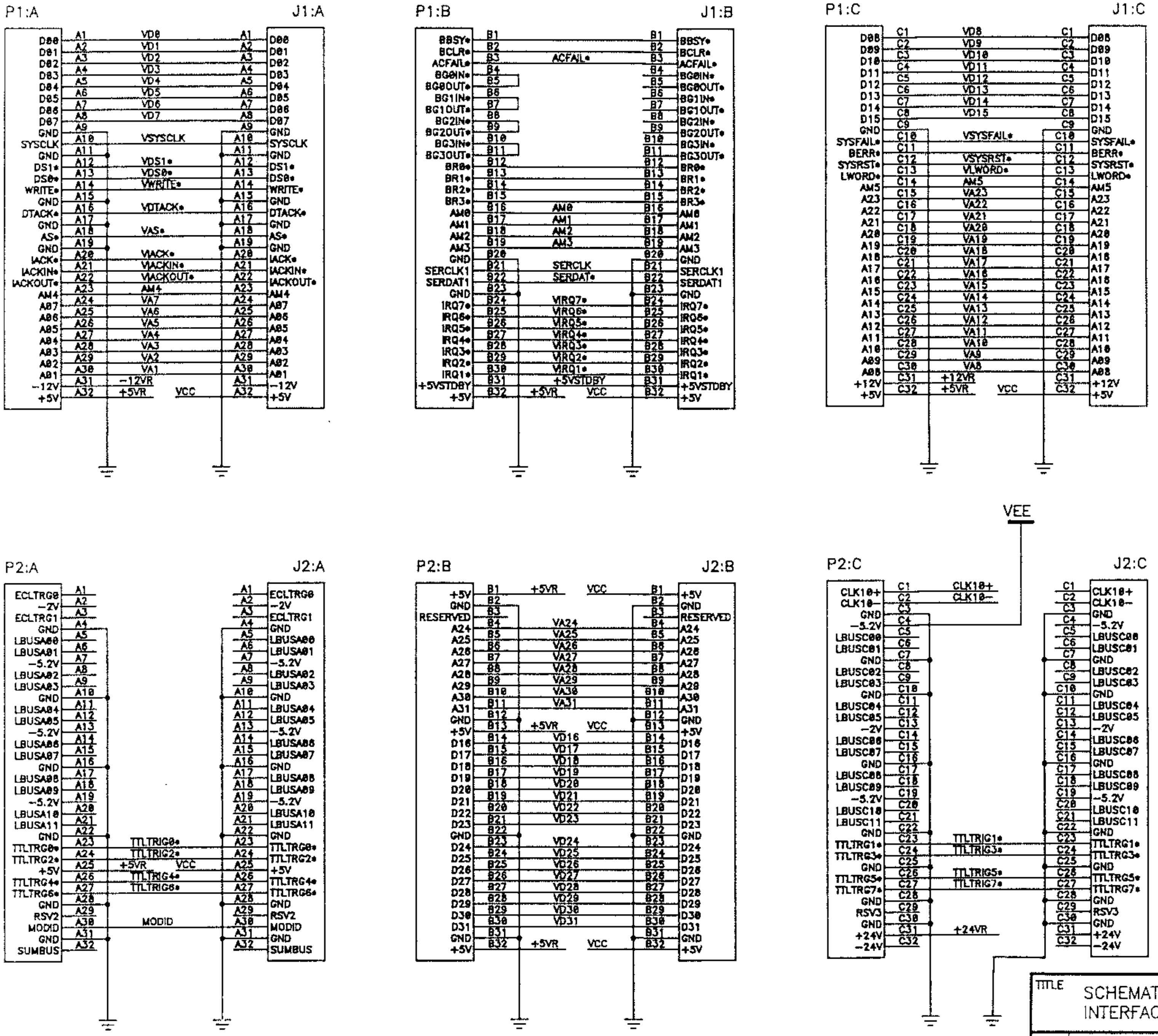
TITLE SCHEMATIC, 8 CHANNEL VXI SERIAL INTERFACE

SIZE C | CAGE CODE | DWG NO. 50-0031-000 | REV. A

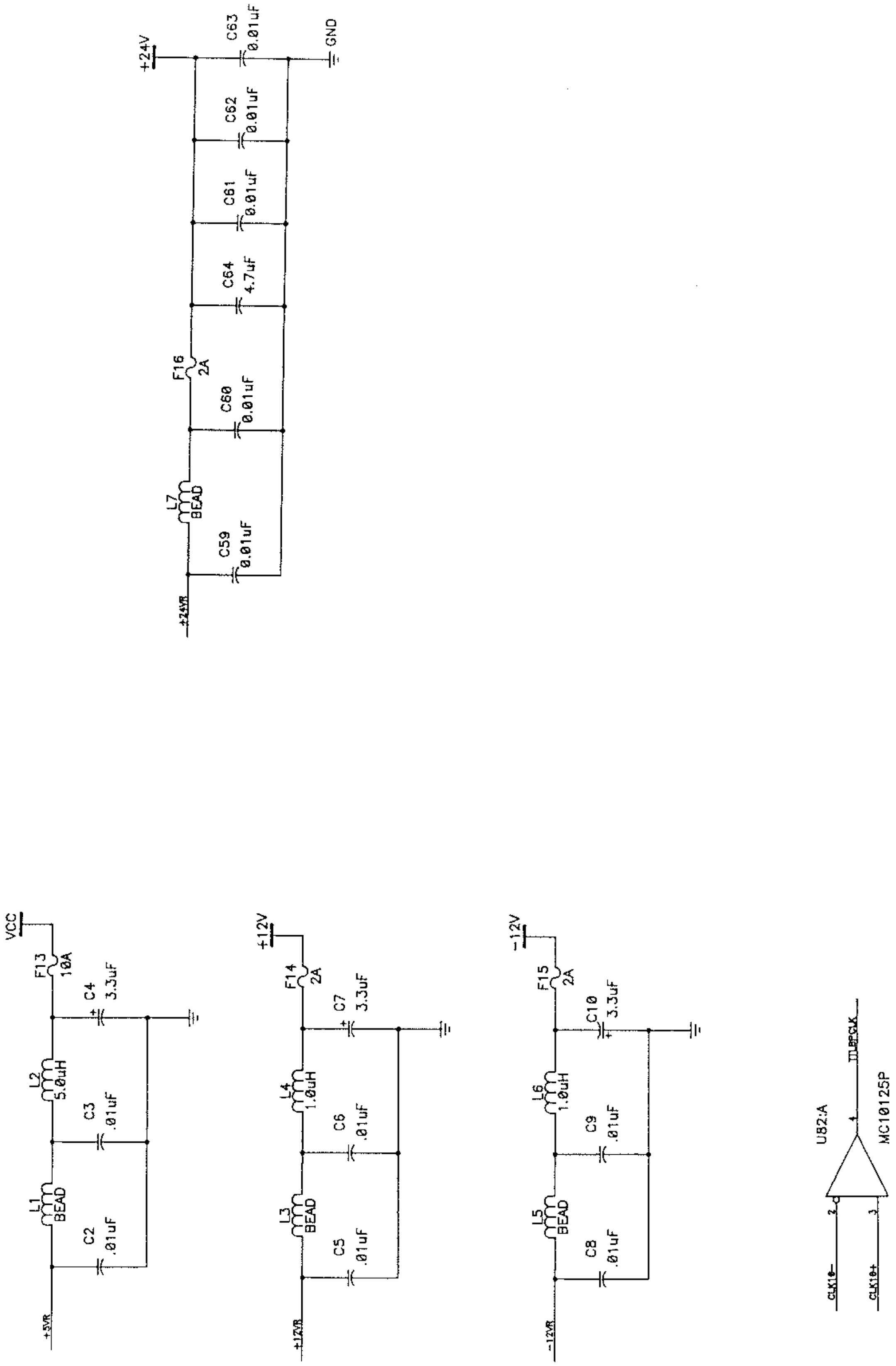
SCALE NONE | FILE 0031-000.S10 | SHEET 18 OF 21



TITLE	SCHEMATIC, 8 CHANNEL VXI SERIAL INTERFACE		
SIZE C	CAGE CODE	DWG NO. 50-0031-000	REV. A
SCALE NONE	FILE 0031-000.S10	SHEET 19 OF 21	


P1:A
J1:A
P1:B
J1:B
P1:C
J1:C
P2:A
J2:A
P2:B
J2:B
P2:C
J2:C
VEE
VD0 VD1 VD2 VD3 VD4 VD5 VD6 VD7
VSYSCLK
VDS1• VDS0• VWRITE•
VDTACK•
VAS•
VIACK• VIACKIN• VIACKOUT•
AM4 VA7 VA6 VA5 VA4 VA3 VA2 VA1
−12VR
+5VR
VCC
ACFAIL•
AM0 AM1 AM2 AM3
SERCLK SERDAT•
VIRQ7• VIRQ6• VIRQ5• VIRQ4• VIRQ3• VIRQ2• VIRQ1•
+5VSTDBY
VD8 VD9 VD10 VD11 VD12 VD13 VD14 VD15
VSYSFAIL•
VSYSRST• VLWORD•
AM5 VA23 VA22 VA21 VA20 VA19 VA18 VA17 VA16 VA15 VA14 VA13 VA12 VA11 VA10 VA9 VA8
+12VR
TTLTRIG0• TTLTRIG2• TTLTRIG4• TTLTRIG6•
MODID
VA24 VA25 VA26 VA27 VA28 VA29 VA30 VA31
VD16 VD17 VD18 VD19 VD20 VD21 VD22 VD23
VD24 VD25 VD26 VD27 VD28 VD29 VD30 VD31
CLK10+ CLK10−
TTLTRIG1• TTLTRIG3• TTLTRIG5• TTLTRIG7•
+24VR

TITLE	SCHEMATIC, 8 CHANNEL VXI SERIAL INTERFACE			
SIZE C	CAGE CODE	DWG NO. 50−0031−000		REV. A
SCALE NONE	FILE 0031−000.S11		SHEET 20 OF 21	



TITLE	SCHEMATIC, 8 CHANNEL VXI SERIAL INTERFACE		
SIZE C	CAGE CODE	DWG NO. 50-0031-000	REV. A
SCALE NONE	FILE 0031-000.S11	SHEET 21 OF 21	

TITLE: FINAL ASSY, 4 CHANNEL VXI SERIAL INTERFACE
PARTNUMBER: 70-0036-000
REVISION: A

ITEM	PART NUMBER	DESCRIPITION	QTY	REFERENCE
1	19-0010-011	IC, EPROM, SC6065 FIRMWARE	1	
2	27-0056-002	CONN, PIN STRIP, 2 PIN, .156 HIGH	2	
3	27-0056-032	CONN, PIN STRIP, 32 PIN, .156 HIGH	2	
4				
5	37-0010-025	SCREW, 4-40 PAN HEAD, SEMS	5	
6	37-0011-062	SCREW, 4-40 PAN HEAD	4	
7	37-0012-004	WASHER, FLAT, 4-40	4	
8	37-0013-004	WASHER, SPLIT LOCK, 4-40	4	
9				
10	37-0017-000	VXI EXTRACTOR HANDLE, TOP, GREY	1	
11	37-0017-001	VXI EXTRACTOR HANDLE, BOTTOM, GREY	1	
12	37-0017-002	VXI EXTRACTOR HARDWARE KIT	1	
13	37-0026-000	JACK SCREW, 4-40 X 0.13"	4	
14	37-0047-025	SCREW, 4-40 X 1/4" FLAT HEAD UNDERCUT, ZINC	12	
15	37-0050-018	SCREW, 2-56 X 3/16" FLAT HEAD PHIL, STAINLESS	6	
16				
17	41-0010-002	BASE PLATE, C SIZE BREADBOARD, SINGLE	1	
18	41-0010-003	COVER PLATE, C SIZE BREADBOARD, SINGLE	1	
19	41-0010-016	FRONT PANEL, 4 CHANNEL SC6065 SERIAL INTERFACE	1	
20				
21	43-0010-000	LABEL, STATIC WARNING	1	
22	43-0018-000	LABEL, RDII SERIAL/MODEL NUMBER FOR VXI PROD.	1	
23				
24	52-0011-000	ASSY, PCB, VXI INTERFACE MODULE, 8 MHz	1	
25	52-0033-000	ASSY, PCB, 4 CHANNEL VXI SERIAL INTERFACE	1	
26				

TITLE: FINAL ASSY, 8 CHANNEL VXI SERIAL INTERFACE
PARTNUMBER: 70-0036-001
REVISION: A

ITEM	PART NUMBER	DESCRIPITION	QTY	REFERENCE
1	19-0010-011	IC, EPROM, SC6065 FIRMWARE	1	
2	27-0056-002	CONN, PIN STRIP, 2 PIN, .156 HIGH	2	
3	27-0056-032	CONN, PIN STRIP, 32 PIN, .156 HIGH	2	
4				
5	37-0010-025	SCREW, 4-40 PAN HEAD, SEMS	5	
6	37-0011-062	SCREW, 4-40 PAN HEAD	4	
7	37-0012-004	WASHER, FLAT, 4-40	4	
8	37-0013-004	WASHER, SPLIT LOCK, 4-40	4	
9				
10	37-0017-000	VXI EXTRACTOR HANDLE, TOP, GREY	1	
11	37-0017-001	VXI EXTRACTOR HANDLE, BOTTOM, GREY	1	
12	37-0017-002	VXI EXTRACTOR HARDWARE KIT	1	
13	37-0026-000	JACK SCREW, 4-40 X 0.13"	8	
14	37-0047-025	SCREW, 4-40 X 1/4" FLAT HEAD UNDERCUT, ZINC	12	
15	37-0050-018	SCREW, 2-56 X 3/16" FLAT HEAD PHIL, STAINLESS	6	
16				
17	41-0010-002	BASE PLATE, C SIZE BREADBOARD, SINGLE	1	
18	41-0010-003	COVER PLATE, C SIZE BREADBOARD, SINGLE	1	
19	41-0010-017	FRONT PANEL, 8 CHANNEL SC6065 SERIAL INTERFACE	1	
20				
21	43-0010-000	LABEL, STATIC WARNING	1	
22	43-0018-000	LABEL, RDII SERIAL/MODEL NUMBER FOR VXI PROD.	1	
23				
24	52-0011-001	ASSY, PCB, VXI INTERFACE MODULE, 12.5 MHz	1	
25	52-0033-001	ASSY, PCB, 8 CHANNEL VXI SERIAL INTERFACE	1	
26				

PARTLIST, VXI INTERFACE MODULE, 12.5 MHz, P/N 52-0011-001 REVISION B

ITEM	P/N	DESCRIPTION	QTY	REFERENCE
1	01-0011-102	RES, CHIP, 1K 5%, 0.06W, SMD 0805	10	R5,6,19,21,26,27,38-41
2	01-0011-103	RES, CHIP, 10K 5%, 0.06W, SMD 0805	6	R1-4
3	01-0011-220	RES, CHIP, 22 5%, 0.06W, SMD 0805	10	R42-51
4	01-0011-331	RES, CHIP, 330 5%, 0.06W, SMD 0805	3	R23,24,52
5	01-0011-471	RES, CHIP, 470 5%, 0.06W, SMD 0805	2	R20,53
6	01-0011-472	RES, CHIP, 4.7K 5%, 0.06W, SMD 0805	20	R7-16,22,28,30-37,56,57
7	01-0013-825	RES, CHIP, 825 1%, 0.06W, SMD 0805	1	R29
8	05-0010-103	CAP, MONO, 0.01 uF 50V, SMD 1206	26	C1,3-27
9	05-0012-336	CAP, TANT, 3.3 uF 16V, SMD	2	C2,28
10	15-0010-005	IC, INT, TL7705, SO8	1	U23
11	17-0010-032	IC, DIG, 74LS32, SO14	1	U24
12	17-0010-148	IC, DIG, 74LS148, SO16	1	U14
13	17-0011-000	IC, DIG, 74F00, SO14	1	U35
14	17-0011-032	IC, DIG, 74F32, SO14	1	U39
15	17-0011-074	IC, DIG, 74F74, SO14	1	U7
16	17-0011-138	IC, DIG, 74F138, SO16	2	U9,34
17	17-0011-245	IC, DIG, 74F245, SO20	1	U47
18	17-0011-367	IC, DIG, 74F367, SO16	1	U46
19	17-0011-652	IC, DIG, 74F652, SO24	2	U37,38
20	17-0011-641	IC, DIG, 74F641, SO20	2	U21,22
21	17-0012-008	IC, DIG, 74HCT08, SO14	1	U11
22	17-0012-030	IC, DIG, 74HCT30, SO14	1	U42
23	17-0012-074	IC, DIG, 74HCT74, SO14	2	U20,31
24	17-0012-138	IC, DIG, 74HCT138, SO16	2	U16,17
25	17-0012-151	IC, DIG, 74HCT151, SO16	1	U15
26	17-0012-244	IC, DIG, 74HCT244, SO20	3	U27,43,44
27	17-0012-374	IC, DIG, 74HCT374, SO20	2	U30,41
28	17-0012-367	IC, DIG, 74HCT367, SO16	2	U32,33
29	17-0012-688	IC, DIG, 74HCT688, SO20	2	U18,26
30	17-0013-032	IC, DIG, 74HC32, SO14	1	U4
31	17-0014-038	IC, DIG, 74S38, SO14	1	U8
32	17-0015-001	IC, MICROPROCESSOR, MC68000FN-12	1	U10
33	17-0016-000	IC, TIMER, MC68230FN-8	1	U19
34				
35	19-0011-000	IC, MEM, RAM, KM62256ALG-10	2	U2,3
36	19-0012-000	IC, MEM, EEPROM, NQ28C64-150	2	U5,6
37	19-0014-000	IC, MEM, DUAL PORT RAM, IDT7130SAJ-100	2	U28,29
38	21-0010-001	IC, PAL, 22V10FN-25	1	U36
39	21-0011-001	IC, PAL, 16L8AFN-15	1	U40
40	21-0012-001	IC, PAL, 16R4AFN-15	1	U45
41	21-0013-001	IC, PAL, 20L8AFN-25	1	U12
42	27-0056-032	CONN, PIN STRIP, 32 PIN, .156 HIGH	5	P101A,B,C P102A,C
43	27-0013-004	CONN, RIGHT ANGLE, 4 PINS	1	P1
44				
45	27-0014-044	SOCKET, PLCC, 44 PINS	1	W/U1
46	27-0014-068	SOCKET, PLCC, 68 PINS	1	W/U10
47	29-0011-008	SWITCH, DIP, 8 POSITIONS, SMD	1	SW1
48	33-0011-000	PCB, VXI INTERFACE MODULE	1	
49	37-0014-031	SPACER, SWAGE, 4-40 X 5/16", 3/16 OD	4	
50	50-0011-000	SCHEMATIC, VXI INTERFACE MODULE		REFERENCE
51	13-0010-000	TRAN, NPN, MMBT3904, SMD	2	Q1,2
52	15-0011-000	IC, TIMER, NE555, SMD	1	U25
53	01-0011-104	RES, CHIP, 100K 5%, 0.06W, SMD 0805	1	R25
54	01-0018-102	RES, NETWORK, SMD, 8 RES, 1K	1	Z1
55				
56	37-0012-250	OSCILLATOR, 25.0 MHz TTL OUT	1	U13

PARTLIST, VXI INTERFACE MODULE, 8 MHz, P/N 52-0011-000 REVISION B

ITEM	P/N	DESCRIPTION	QTY	REFERENCE
1	01-0011-102	RES, CHIP, 1K 5%, 0.06W, SMD 0805	10	R5,6,19,21,26,27,38-41
2	01-0011-103	RES, CHIP, 10K 5%, 0.06W, SMD 0805	6	R1-4
3	01-0011-220	RES, CHIP, 22 5%, 0.06W, SMD 0805	10	R42-51
4	01-0011-331	RES, CHIP, 330 5%, 0.06W, SMD 0805	3	R23,24,52
5	01-0011-471	RES, CHIP, 470 5%, 0.06W, SMD 0805	2	R20,53
6	01-0011-472	RES, CHIP, 4.7K 5%, 0.06W, SMD 0805	20	R7-16,22,28,30-37,56,57
7	01-0013-825	RES, CHIP, 825 1%, 0.06W, SMD 0805	1	R29
8	05-0010-103	CAP, MONO, 0.01 uF 50V, SMD 1206	26	C1,3-27
9	05-0012-336	CAP, TANT, 3.3 uF 16V, SMD	2	C2,28
10	15-0010-005	IC, INT, TL7705, SO8	1	U23
11	17-0010-032	IC, DIG, 74LS32, SO14	1	U24
12	17-0010-148	IC, DIG, 74LS148, SO16	1	U14
13	17-0011-000	IC, DIG, 74F00, SO14	1	U35
14	17-0011-032	IC, DIG, 74F32, SO14	1	U39
15	17-0011-074	IC, DIG, 74F74, SO14	1	U7
16	17-0011-138	IC, DIG, 74F138, SO16	2	U9,34
17	17-0011-245	IC, DIG, 74F245, SO20	1	U47
18	17-0011-367	IC, DIG, 74F367, SO16	1	U46
19	17-0011-652	IC, DIG, 74F652, SO24	2	U37,38
20	17-0011-641	IC, DIG, 74F641, SO20	2	U21,22
21	17-0012-008	IC, DIG, 74HCT08, SO14	1	U11
22	17-0012-030	IC, DIG, 74HCT30, SO14	1	U42
23	17-0012-074	IC, DIG, 74HCT74, SO14	2	U20,31
24	17-0012-138	IC, DIG, 74HCT138, SO16	2	U16,17
25	17-0012-151	IC, DIG, 74HCT151, SO16	1	U15
26	17-0012-244	IC, DIG, 74HCT244, SO20	3	U27,43,44
27	17-0012-374	IC, DIG, 74HCT374, SO20	2	U30,41
28	17-0012-367	IC, DIG, 74HCT367, SO16	2	U32,33
29	17-0012-688	IC, DIG, 74HCT688, SO20	2	U18,26
30	17-0013-032	IC, DIG, 74HC32, SO14	1	U4
31	17-0014-038	IC, DIG, 74S38, SO14	1	U8
32	17-0015-000	IC, MICROPROCESSOR, MC68000FN-8	1	U10
33	17-0016-000	IC, TIMER, MC68230FN-8	1	U19
34				
35	19-0011-000	IC, MEM, RAM, KM62256ALG-10	2	U2,3
36	19-0013-000	IC, MEM, EEPROM, NQ28C64-200	2	U5,6
37	19-0014-000	IC, MEM, DUAL PORT RAM, IDT7130SAJ-100	2	U28,29
38	21-0010-001	IC, PAL, 22V10FN-25	1	U36
39	21-0011-001	IC, PAL, 16L8AFN-15	1	U40
40	21-0012-001	IC, PAL, 16R4AFN-15	1	U45
41	21-0013-001	IC, PAL, 20L8AFN-25	1	U12
42	27-0056-032	CONN, PIN STRIP, 32 PIN, .156 HIGH	5	P101A,B,C P102A,C
43	27-0013-004	CONN, RIGHT ANGLE, 4 PINS	1	P1
44				
45	27-0014-044	SOCKET, PLCC, 44 PINS	1	W/U1
46	27-0014-068	SOCKET, PLCC, 68 PINS	1	W/U10
47	29-0011-008	SWITCH, DIP, 8 POSITIONS, SMD	1	SW1
48	33-0011-000	PCB, VXI INTERFACE MODULE	1	
49	37-0014-031	SPACER, SWAGE, 4-40 X 5/16",, 3/16 OD	4	
50	50-0011-000	SCHEMATIC, VXI INTERFACE MODULE		REFERENCE
51	13-0010-000	TRAN, NPN, MMBT3904, SMD	2	Q1,2
52	15-0011-000	IC, TIMER, NE555, SMD	1	U25
53	01-0011-104	RES, CHIP, 100K 5%, 0.06W, SMD 0805	1	R25
54	01-0018-102	RES, NETWORK, SMD, 8 RES, 1K	1	Z1

PARTLIST 52-0033-000 REVISION A
ASSY, PCB, 4 CHANNEL VXI SERIAL INTERFACE

ITEM	PART NUMBER	DESCRIPTION	QTY	REF. DES.
1	01-0010-103	RES, CHIP, 10K 5%, 0.06W, 0805	6	R28-R33
2	01-0019-103	RES, CARBON COMP, 10K OHM 5% 1/4W	6	R1,R2,R3,R5,R6,R7
3	01-0019-151	RES, CARBON COMP, 150 OHM 5% 1/4W	3	R4,R26,R27
4	01-0019-472	RES, CARBON COMP, 4.7K OHM 5% 1/4W	11	R14-R24
5				
6	05-0011-335	CAP, TANT, 3.3uF 16V 20% RADIAL	5	C4,C7,C10,C21,C64
7	05-0014-103	CAP, MONOLYTHIC, 0.01uF 50V 20%. AXIAL	46	C1,C2,C3,C5,C6,C8,C9,C11-C14, C19,C20,C22-C25,C30-C36,C39, C41,C46,C65,C66
8	09-0011-000	INDUCTOR, BEAD ON WIRE	4	L1,L3,L5,L7
9	09-0014-500	INDUCTOR, POWER, RADIAL 5uH 10A	1	L2
10				
11	11-0018-001	LED, PCB MOUNT, T-1 RIGHT ANGLE, RED	1	D1
12	11-0018-002	LED, PCB MOUNT, T-1 RIGHT ANGLE, GREEN	1	D2
13				
14	17-0023-000	IC, DUAL UART, SCN2681, 44 PIN PLCC	2	U2,U3
15	17-0024-004	IC, HEX INVERTER, 74HCT04, PLASTIC DIP	1	U81
16	17-0024-021	IC, DUAL 4 INPUT AND GATE, 74HCT21, PLACTIC DIP	1	U71
17	17-0024-138	IC, 3 TO 8 DECODER, 74HCT138, PLASTIC DIP	2	U68,U69
18	17-0024-244	IC, OCTAL TRI-STATE BUFFER, 74HCT244, PLASTIC DIP	3	U50,U53,U56
19	17-0024-245	IC, OCTAL BI-DIRECTIONAL BUFFER, 74HCT245, PDIP	2	U59,U62
20	17-0024-273	IC, OCTAL REGISTER W/RESET, 74HCT273, PLASTIC DIP	5	U51,U54,U57,U70,U72
21				
22	19-0022-080	IC, RAM, 32K BY 8, 80nS, PLASTIC DIP	4	U73-U76
23				
24	21-0016-003	IC, PAL, PROG, 22V10, SC6065 INTERFACE, U65	1	U65
25	21-0019-001	IC, PAL, PROG, 16L8, SC6065 RAM DECODER, U78	1	U78
26	21-0019-002	IC, PAL, PROG, 16L8, SC6065 XMIT DECODER	2	U16,U27
27	21-0020-001	IC, PAL, PROG, 20L8, SC6065 RECEIVE DECODER	2	U12,U23
28				
29	23-0012-000	IC, INTERFACE, RELAY DRIVER, ULN-2803A, PDIP	3	U52,U55,U58
30	23-0017-000	IC, INTERFACE, QUAD RS-422 DRIVER, 26LS31, PDIP	4	U8,U10,U19,U21
31	23-0023-000	IC, INTERFACE, QUAD RS-422 RECEIVER, 26LS32, PDIP	4	U9,U14,U20,U25
32	23-0024-000	IC, INTERFACE, QUAD RS-232 RECEIVER, SN75189, PDIP	4	U7,U11,U18,U22
33	23-0025-000	IC, INTERFACE, DUAL RS-232/423 DRIVER, UA9636, PDIP	6	U6,U13,U15,U17,U24,U26
34				
35				
36	27-0010-096	CONN, EURO-DIN, 96 PIN MALE RIGHT ANGLE	2	P1,P2
37	27-0012-004	CONN, SOCKET STRIP, 2 PIN	2	P9 (2 PLACES)
38	27-0012-032	CONN, SOCKET STRIP, 32 PIN	7	J1A,J1B,J1C,J2A,J2C,P7(2 PLCS)
39	27-0014-044	SOCKET, PLCC, 44 PIN	2	XU2,XU3
40	27-0054-026	CONN, HIGH DENSITY DSUB, 26 PIN F/M, PC MOUNT	2	J100,J101
41	27-0028-008	SOCKET, LOW PROFILE DIP, 8 PIN	6	XU6,XU13,XU15,XU17,XU24,XU26
42	27-0028-014	SOCKET, LOW PROFILE DIP, 14 PIN	4	XU7,XU11,XU18,XU22
43	27-0028-016	SOCKET, LOW PROFILE DIP, 16 PIN	1 8	XU8,XU9,XU10,XU14,XU19,XU20,XU21,XU25
44				
45				
46	33-0030-000	PCB, 8 CHANNEL VXI SERIAL INTERFACE	1	
47				
48	35-0010-100	FUSE, AXIAL, 10A	1	F13
49	35-0029-008	FUSE, POLY-FUSE SELF HEALING	4	F1,F4,F14,F15
50	35-0012-036	OSCILLATOR, 3.6864 MHz TTL OUTPUT	1	U1
51	35-0013-024	RELAY, 1A DPDT, PC MOUNT TQ2E-24V	24	K1-K24
52	35-0028-000	WIRE, 24 GA SOLID BUS, COPPER/TIN PLATE	A/R	

PARTLIST 52-0033-001 REVISION A

ASSY, PCB, 8 CHANNEL VXI SERIAL INTERFACE

ITEM	PART NUMBER	DESCRIPTION	QTY	REF. DES.
1	01-0010-103	RES, CHIP, 10K 5%, 0.06W, 0805	12	R28-R39
2	01-0019-103	RES, CARBON COMP, 10K OHM 5% 1/4W	12	R1,R2,R3,R5-R13
3	01-0019-151	RES, CARBON COMP, 150 OHM 5% 1/4W	3	R4,R26,R27
4	01-0019-472	RES, CARBON COMP, 4.7K OHM 5% 1/4W	11	R14-R24
5				
6	05-0011-335	CAP, TANT, 3.3uF 16V 20% RADIAL	5	C4,C7,C10,C21,C64
7	05-0014-103	CAP, MONOLYTHIC, 0.01uF 50V 20%. AXIAL	61	C1,C2,C3,C5,C6,C8,C9,C11-C20,C22-C63,C65,C66
8	09-0011-000	INDUCTOR, BEAD ON WIRE	4	L1,L3,L5,L7
9	09-0014-500	INDUCTOR, POWER, RADIAL 5uH 10A	1	L2
10				
11	11-0018-001	LED, PCB MOUNT, T-1 RIGHT ANGLE, RED	1	D1
12	11-0018-002	LED, PCB MOUNT, T-1 RIGHT ANGLE, GREEN	1	D2
13				
14	17-0023-000	IC, DUAL UART, SCN2681, 44 PIN PLCC	4	U2-U5
15	17-0024-004	IC, HEX INVERTER, 74HCT04, PLASTIC DIP	1	U81
16	17-0024-021	IC, DUAL 4 INPUT AND GATE, 74HCT21, PLACTIC DIP	1	U71
17	17-0024-138	IC, 3 TO 8 DECODER, 74HCT138, PLASTIC DIP	2	U68,U69
18	17-0024-244	IC, OCTAL TRI-STATE BUFFER, 74HCT244, PLASTIC DIP	3	U50,U53,U56
19	17-0024-245	IC, OCTAL BI-DIRECTIONAL BUFFER, 74HCT245, PDIP	2	U59,U62
20	17-0024-273	IC, OCTAL REGISTER W/RESET, 74HCT273, PLASTIC DIP	8	U51,U54,U57,U60,U63,U66,U70,U72
21				
22	19-0022-080	IC, RAM, 32K BY 8, 80nS, PLASTIC DIP	4	U73-U76
23				
24	21-0016-003	IC, PAL, PROG, 22V10, SC6065 INTERFACE, U65	1	U65
25	21-0019-001	IC, PAL, PROG, 16L8, SC6065 RAM DECODER, U78	1	U78
26	21-0019-002	IC, PAL, PROG, 16L8, SC6065 XMIT DECODER	4	U16,U27,U38,U49
27	21-0020-001	IC, PAL, PROG, 20L8, SC6065 RECEIVE DECODER	4	U12,U23,U34,U45
28				
29	23-0012-000	IC, INTERFACE, RELAY DRIVER, ULN-2803A, PDIP	6	U52,U55,U58,U61,U64,U67
30	23-0017-000	IC, INTERFACE, QUAD RS-422 DRIVER, 26LS31, PDIP	8	U8,U10,U19,U21,U30,U32,U41,U43
31	23-0023-000	IC, INTERFACE, QUAD RS-422 RECEIVER, 26LS32, PDIP	8	U9,U14,U20,U25,U31,U36,U42,U47
32	23-0024-000	IC, INTERFACE, QUAD RS-232 RECEIVER, SN75189, PDIP	8	U7,U11,U18,U22,U29,U33,U40,U44
33	23-0025-000	IC, INTERFACE, DUAL RS-232/423 DRIVER, UA9636, PDIP	12	U6,U13,U15,U17,U24,U26,U28,U35, U37,U39,U46,U48
34				
35				
36	27-0010-096	CONN, EURO-DIN, 96 PIN MALE RIGHT ANGLE	2	P1,P2
37	27-0012-004	CONN, SOCKET STRIP, 2 PIN	2	P9 (2 PLACES)
38	27-0012-032	CONN, SOCKET STRIP, 32 PIN	7	J1A,J1B,J1C,J2A,J2C,P7(2 PLCS)
39	27-0014-044	SOCKET, PLCC, 44 PIN	4	XU2,XU3,XU4,XU5
40	27-0054-026	CONN, HIGH DENSITY DSUB, 26 PIN F/M, PC MOUNT	4	J100,J101,J102,J103
41	27-0028-008	SOCKET, LOW PROFILE DIP, 8 PIN	12	XU6,XU13,XU15,XU17,XU24,XU26,XU28,XU35,XU37, XU39,XU46,XU48
42	27-0028-014	SOCKET, LOW PROFILE DIP, 14 PIN	8	XU7,XU11,XU18,XU22,XU29,XU33,XU40,XU44
43	27-0028-016	SOCKET, LOW PROFILE DIP, 16 PIN	16	XU8,XU9,XU10,XU14,XU19,XU20,XU21,XU25,XU30, XU31,XU32,XU36,XU41,XU42,XU43,XU47
44				
45				
46	33-0030-000	PCB, 8 CHANNEL VXI SERIAL INTERFACE	1	
47				
48	35-0010-100	FUSE, AXIAL, 10A	1	F13
49	35-0029-008	FUSE, POLY-FUSE SELF HEALING	6	F1,F4,F7,F10,F14,F15
50	35-0012-036	OSCILLATOR, 3.6864 MHz TTL OUTPUT	1	U1
51	35-0013-024	RELAY, 1A DPDT, PC MOUNT TQ2E-24V	48	K1-K48
52	35-0028-000	WIRE, 24 GA SOLID BUS, COPPER/TIN PLATE	A/R	

The following harness assemblies are used to connect Racal Instruments Model 6065-4 to Freedom Series Test Receiver Interfaces.

Each harness documentation consists of an assembly drawing, parts list, system wire list and wire list.

407316	Virginia Panel, Inc. Series VP90 Interface Harness
407317	TTI Testron Inc. Interface Harness

For more information on Racal Instruments complete line of Test Receiver Interface solutions, contact your Sales Representative.

DWG. NO. 407316 SH 1 REV. A

REVISIONS				
ZONE	REV	DESCRIPTION	DATE	APPROVED
		RELEASED PER DRN NO.		
		REVISED PER EO NO.		
		REVISED PER EO NO.		
		REVISED PER EO NO.		

NOTES:

1. SEE WIRE LIST FOR CONNECTIONS AND CONTACT ASSIGMENT.

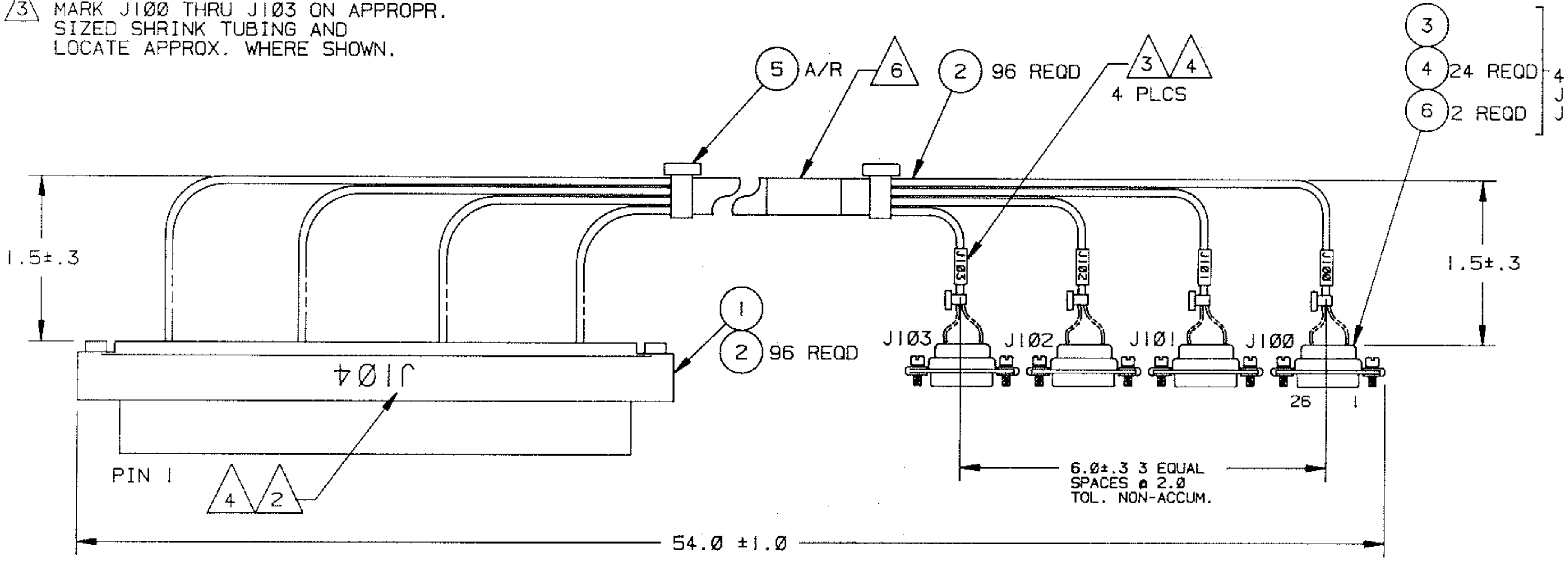
△2 WHITE INK STAMP J104 AT THE SIDE OF CONNECTOR AND LOCATE APPROX. WHERE SHOWN .

△3 MARK J100 THRU J103 ON APPROPR. SIZED SHRINK TUBING AND LOCATE APPROX. WHERE SHOWN.

△4 SPRAY PROTECTIVE COATING (ITEM 7) AFTER MARKING.

5. MATING ITA CONNECTOR FOR J104 IS VIRGINIA PANEL P/N 510 108 126. USE ITA PIN P/N 610 110 108.

△6 MARK 407316 ON APPROPRIATE SIZED SHRINK TUBING APPROXIMATELY WHERE SHOWN. PLACE APPROPRIATE SIZED CLEAR SHRINK TUBING OVER MARKING.



PRECIOUS METAL INDICATOR CODE								
WEIGHTS								
STRESS								
MATERIALS & PROCESSES								
QUALITY ASSURANCE	7/20/95							FREEDOM
MANUFACTURING ENGRG J. Villacorta	2/17/95	PART DASH	NEXT	FINAL		NEXT ASSY		USED ON
ADDITIONAL APPROVALS	DATE	NO.	QTY REQ'D PER ASSY			APPLICATION		

UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES TOLERANCES ARE :

ANGLES	DECIMALS	HOLE Ø
MACH'D ± 0°30'	.XX ± .02	+.004
FORMED ± 1°00'	.XXX ± .010	-.001

DO NOT SCALE DRAWING

MATERIAL:

FINISH:

SIMILAR TO | SPECIAL MARKING SYM.

CONTRACT NO.

APPROVALS		DATE
DRAWN	KHANH	94-09-07
CHECKED		95-2-7
ENGR		95-2-7
DESIGN ACTIVITY		

RACAL Instruments, Inc.
4 Goodyear St., Irvine, CA. 92718-2002

TITLE
HARNESS ASSY, 6065, VP90

SIZE	CAGE CODE	DWG NO.	REV.
B	21793	407316	A
SCALE NONE	CALC.WT	ACT.WT	SHEET 1 OF 7

980340 REV. F

ENGINEERING PARTS LIST

ITEM	BIN	PART NO.	DESCRIPTION	QTY	REFERENCE
1		602201-003	CON-RCV-PLG096.---T-VP90	1	J104
2		602201-806	PATCHCORD, SIGNAL, 24AWG,60IN	96	W/J104
3		602222	CON-D/S-PLG026C	4	J100-J103
4		601702-221	CONTACT,CONN,PIN	96	W/J100-J103
5		610777	TIE-CA-LKG-.062-.075	A/R	
6		610846	LOCK SCREW ASSY	8	W/J100-J103
7		910541	POLYURETHANE CONF. COAT	A/R	

DOC. NO. 407316

RACAL Instruments, Inc., 4 Goodyear St., Irvine, CA 92718

DOCUMENT TITLE	SIZE	CODE NO.	DOCUMENT NO.	REV
HARNESS ASSEMBLY, 6065, VP90	A	21793	407316	A
	DRN			SHEET 2 of 7

ENGINEERING WIRE LIST

WIRE	FROM	TO	TYPE	PART #	WIRE LEN	REFERENCE
	BLK AA (J104)	Uxx-SLOT yy (J100-J103)	CABLE	407316		SYSTEM WIRE LIST

This system wirelist serves as a template for incorporating this harness assembly into the overall system wirelist. It does not in any way affect the fabrication of this harness assembly.

DOC. NO. 407316

RACAL Instruments, Inc., 4 Goodyear St., Irvine, CA 92718

DOCUMENT TITLE	SIZE	CODE NO.	DOCUMENT NO.	REV
HARNESS ASSEMBLY, 6065, VP90	A	21793	407316	A
	DRN			SHEET 3 of 7

ENGINEERING WIRE LIST

WIRE	FROM	TO	TYPE	PART #	WIRE LEN	REFERENCE
1	J104-33 (602201-003)	J100-1 601702-221	24AWG WHT	602201- 806	54"	GND
2	J104-1 (602201-003)	J100-2 601702-221	24AWG WHT	602201- 806	54"	TXD-A+
3	J104-65 (602201-003)	J100-3 601702-221	24AWG WHT	602201- 806	54"	TXD-A-
4	J104-34 (602201-003)	J100-4 601702-221	24AWG WHT	602201- 806	54"	RXD-A+
5	J104-2 (602201-003)	J100-5 601702-221	24AWG WHT	602201- 806	54"	RXD-A-
6	J104-66 (602201-003)	J100-6 601702-221	24AWG WHT	602201- 806	54"	RTS-A
7	J104-35 (602201-003)	J100-7 601702-221	24AWG WHT	602201- 806	54"	CTS-A
8	J104-3 (602201-003)	J100-8 601702-221	24AWG WHT	602201- 806	54"	DSR-A
9	J104-67 (602201-003)	J100-9 601702-221	24AWG WHT	602201- 806	54"	DCD-A
10	J104-36 (602201-003)	J100-10 601702-221	24AWG WHT	602201- 806	54"	GND
11	J104-4 (602201-003)	J100-11 601702-221	24AWG WHT	602201- 806	54"	TXD-B+
12	J104-68 (602201-003)	J100-12 601702-221	24AWG WHT	602201- 806	54"	TXD-B-
13	J104-37 (602201-003)	J100-13 601702-221	24AWG WHT	602201- 806	54"	RXD-B+
14	J104-5 (602201-003)	J100-14 601702-221	24AWG WHT	602201- 806	54"	RXD-B-
15	J104-69 (602201-003)	J100-15 601702-221	24AWG WHT	602201- 806	54"	RTS-B
16	J104-38 (602201-003)	J100-16 601702-221	24AWG WHT	602201- 806	54"	CTS-B
17	J104-6 (602201-003)	J100-17 601702-221	24AWG WHT	602201- 806	54"	DSR-B
18	J104-70 (602201-003)	J100-18 601702-221	24AWG WHT	602201- 806	54"	DCD-B
19	J104-39 (602201-003)	J100-19 601702-221	24AWG WHT	602201- 806	54"	GND
20	J104-7 (602201-003)	J100-20 601702-221	24AWG WHT	602201- 806	54"	+5V
21	J104-71 (602201-003)	J100-21 601702-221	24AWG WHT	602201- 806	54"	+12V
22	J104-40 (602201-003)	J100-22 601702-221	24AWG WHT	602201- 806	54"	-12V
23	J104-8 (602201-003)	J100-23 601702-221	24AWG WHT	602201- 806	54"	GND
24	J104-72 (602201-003)	J100-24 601702-221	24AWG WHT	602201- 806	54"	GND

RACAL Instruments, Inc., 4 Goodyear St., Irvine, CA 92718

DOCUMENT TITLE	SIZE	CODE NO.	DOCUMENT NO.	REV
HARNESS ASSEMBLY, 6065, VP90	**A**	**21793**	**407316**	**A**
	DRN			SHEET 4 of 7

ENGINEERING WIRE LIST

WIRE	FROM	TO	TYPE	PART #	WIRE LEN	REFERENCE
25	J104-41 (602201-003)	J101-20 601702-221	24AWG WHT	602201- 806	54"	+5V
26	J104-9 (602201-003)	J101-21 601702-221	24AWG WHT	602201- 806	54"	+12V
27	J104-73 (602201-003)	J101-22 601702-221	24AWG WHT	602201- 806	54"	-12V
28	J104-42 (602201-003)	J101-23 601702-221	24AWG WHT	602201- 806	54"	GND
29	J104-10 (602201-003)	J101-1 601702-221	24AWG WHT	602201- 806	54"	GND
30	J104-74 (602201-003)	J101-2 601702-221	24AWG WHT	602201- 806	54"	TXD-C+
31	J104-43 (602201-003)	J101-3 601702-221	24AWG WHT	602201- 806	54"	TXD-C-
32	J104-11 (602201-003)	J101-4 601702-221	24AWG WHT	602201- 806	54"	RXD-C+
33	J104-75 (602201-003)	J101-5 601702-221	24AWG WHT	602201- 806	54"	RXD-C-
34	J104-44 (602201-003)	J101-6 601702-221	24AWG WHT	602201- 806	54"	RTS-C
35	J104-12 (602201-003)	J101-7 601702-221	24AWG WHT	602201- 806	54"	DTS-C
36	J104-76 (602201-003)	J101-8 601702-221	24AWG WHT	602201- 806	54"	DSR-C
37	J104-45 (602201-003)	J101-9 601702-221	24AWG WHT	602201- 806	54"	DCD-C
38	J104-13 (602201-003)	J101-10 601702-221	24AWG WHT	602201- 806	54"	GND
39	J104-77 (602201-003)	J101-11 601702-221	24AWG WHT	602201- 806	54"	TXD-D+
40	J104-46 (602201-003)	J101-12 601702-221	24AWG WHT	602201- 806	54"	TXD-D-
41	J104-14 (602201-003)	J101-13 601702-221	24AWG WHT	602201- 806	54"	RXD-D+
42	J104-78 (602201-003)	J101-14 601702-221	24AWG WHT	602201- 806	54"	RXD-D-
43	J104-47 (602201-003)	J101-15 601702-221	24AWG WHT	602201- 806	54"	RTS-D
44	J104-15 (602201-003)	J101-16 601702-221	24AWG WHT	602201- 806	54"	DTS-D
45	J104-79 (602201-003)	J101-17 601702-221	24AWG WHT	602201- 806	54"	DSR-D
46	J104-48 (602201-003)	J101-18 601702-221	24AWG WHT	602201- 806	54"	DCD-D
47	J104-16 (602201-003)	J101-19 601702-221	24AWG WHT	602201- 806	54"	GND
48	J104-80 (602201-003)	J101-24 601702-221	24AWG WHT	602201- 806	54"	GND

DOC. NO. 407316

RACAL Instruments, Inc., 4 Goodyear St., Irvine, CA 92718

DOCUMENT TITLE	SIZE	CODE NO.	DOCUMENT NO.	REV
HARNESS ASSEMBLY, 6065, VP90	A	21793	407316	A
	DRN			SHEET 5 of 7

ENGINEERING WIRE LIST

WIRE	FROM	TO	TYPE	PART #	WIRE LEN	REFERENCE
49	J104-49 (602201-003)	J102-1 601702-221	24AWG WHT	602201- 806	54"	GND
50	J104-17 (602201-003)	J102-2 601702-221	24AWG WHT	602201- 806	54"	TXD-E+
51	J104-81 (602201-003)	J102-3 601702-221	24AWG WHT	602201- 806	54"	TXD-E-
52	J104-50 (602201-003)	J102-4 601702-221	24AWG WHT	602201- 806	54"	RXD-E+
53	J104-18 (602201-003)	J102-5 601702-221	24AWG WHT	602201- 806	54"	RXD-E-
54	J104-82 (602201-003)	J102-6 601702-221	24AWG WHT	602201- 806	54"	RTS-E
55	J104-51 (602201-003)	J102-7 601702-221	24AWG WHT	602201- 806	54"	CTS-E
56	J104-19 (602201-003)	J102-8 601702-221	24AWG WHT	602201- 806	54"	DSR-E
57	J104-83 (602201-003)	J102-9 601702-221	24AWG WHT	602201- 806	54"	DCD-E
58	J104-52 (602201-003)	J102-10 601702-221	24AWG WHT	602201- 806	54"	GND
59	J104-20 (602201-003)	J102-11 601702-221	24AWG WHT	602201- 806	54"	TXD-F+
60	J104-84 (602201-003)	J102-12 601702-221	24AWG WHT	602201- 806	54"	TXD-F-
61	J104-53 (602201-003)	J102-13 601702-221	24AWG WHT	602201- 806	54"	RXD-F+
62	J104-21 (602201-003)	J102-14 601702-221	24AWG WHT	602201- 806	54"	RXD-F-
63	J104-85 (602201-003)	J102-15 601702-221	24AWG WHT	602201- 806	54"	RTS-F
64	J104-54 (602201-003)	J102-16 601702-221	24AWG WHT	602201- 806	54"	CTS-F
65	J104-22 (602201-003)	J102-17 601702-221	24AWG WHT	602201- 806	54"	DSR-F
66	J104-86 (602201-003)	J102-18 601702-221	24AWG WHT	602201- 806	54"	DCD-F
67	J104-55 (602201-003)	J102-19 601702-221	24AWG WHT	602201- 806	54"	GND
68	J104-23 (602201-003)	J102-20 601702-221	24AWG WHT	602201- 806	54"	+5V
69	J104-87 (602201-003)	J102-21 601702-221	24AWG WHT	602201- 806	54"	+12V
70	J104-56 (602201-003)	J102-22 601702-221	24AWG WHT	602201- 806	54"	-12V
71	J104-24 (602201-003)	J102-23 601702-221	24AWG WHT	602201- 806	54"	GND
72	J104-88 (602201-003)	J102-24 601702-221	24AWG WHT	602201- 806	54"	GND

DOC. NO. 407316

RACAL Instruments, Inc., 4 Goodyear St., Irvine, CA 92718

DOCUMENT TITLE	SIZE	CODE NO.	DOCUMENT NO.		REV
HARNESS ASSEMBLY, 6065, VP90	**A**	**21793**	**407316**		**A**
	DRN			SHEET 6 of 7	

ENGINEERING WIRE LIST

WIRE	FROM	TO	TYPE	PART #	WIRE LEN	REFERENCE
73	J104-57 (602201-003)	J103-20 601702-221	24AWG WHT	602201- 806	54"	+5V
74	J104-25 (602201-003)	J103-21 601702-221	24AWG WHT	602201- 806	54"	+12V
75	J104-89 (602201-003)	J103-22 601702-221	24AWG WHT	602201- 806	54"	-12V
76	J104-58 (602201-003)	J103-23 601702-221	24AWG WHT	602201- 806	54"	GND
77	J104-26 (602201-003)	J103-1 601702-221	24AWG WHT	602201- 806	54"	GND
78	J104-90 (602201-003)	J103-2 601702-221	24AWG WHT	602201- 806	54"	TXD-G+
79	J104-59 (602201-003)	J103-3 601702-221	24AWG WHT	602201- 806	54"	TXD-G-
80	J104-27 (602201-003)	J103-4 601702-221	24AWG WHT	602201- 806	54"	RXD-G+
81	J104-91 (602201-003)	J103-5 601702-221	24AWG WHT	602201- 806	54"	RXD-G-
82	J104-60 (602201-003)	J103-6 601702-221	24AWG WHT	602201- 806	54"	RTS-G
83	J104-28 (602201-003)	J103-7 601702-221	24AWG WHT	602201- 806	54"	DTS-G
84	J104-92 (602201-003)	J103-8 601702-221	24AWG WHT	602201- 806	54"	DSR-G
85	J104-61 (602201-003)	J103-9 601702-221	24AWG WHT	602201- 806	54"	DCD-G
86	J104-29 (602201-003)	J103-10 601702-221	24AWG WHT	602201- 806	54"	GND
87	J104-93 (602201-003)	J103-11 601702-221	24AWG WHT	602201- 806	54"	TXD-H+
88	J104-62 (602201-003)	J103-12 601702-221	24AWG WHT	602201- 806	54"	TXD-H-
89	J104-30 (602201-003)	J103-13 601702-221	24AWG WHT	602201- 806	54"	RXD-H+
90	J104-94 (602201-003)	J103-14 601702-221	24AWG WHT	602201- 806	54"	RXD-H-
91	J104-63 (602201-003)	J103-15 601702-221	24AWG WHT	602201- 806	54"	RTS-H
92	J104-31 (602201-003)	J103-16 601702-221	24AWG WHT	602201- 806	54"	DTS-H
93	J104-95 (602201-003)	J103-17 601702-221	24AWG WHT	602201- 806	54"	DSR-H
94	J104-64 (602201-003)	J103-18 601702-221	24AWG WHT	602201- 806	54"	DCD-H
95	J104-32 (602201-003)	J103-19 601702-221	24AWG WHT	602201- 806	54"	GND
96	J104-96 (602201-003)	J103-24 601702-221	24AWG WHT	602201- 806	54"	GND

RACAL Instruments, Inc., 4 Goodyear St., Irvine, CA 92718

DOCUMENT TITLE	SIZE	CODE NO.	DOCUMENT NO.	REV
HARNESS ASSEMBLY, 6065, VP90	**A**	**21793**	**407316**	**A**
	DRN		SHEET 7 of 7	

DWG. NO.	SH	REV.
407317	1	A

PROPRIETARY NOTICE

THIS DOCUMENT AND THE TECHNICAL DATA HEREON DISCLOSED ARE PROPRIETARY TO RACAL INSTRUMENTS INC. AND SHALL NOT, WITHOUT THE EXPRESS WRITTEN PERMISSION OF RACAL INSTRUMENTS INC. BE USED, RELEASED OR DISCLOSED IN WHOLE OR IN PART, OR USED TO SOLICIT QUOTATION FROM A COMPETITIVE SOURCE OR USED FOR MANUFACTURE BY ANYONE OTHER THAN RACAL INSTRUMENTS INC. THE INFORMATION HEREON HAS BEEN DEVELOPED AT PRIVATE EXPENSE, AND MAY ONLY BE USED FOR PURPOSES OF ENGINEERING EVALUATION AND FOR INCORPORATION INTO TECHNICAL SPECIFICATIONS AND OTHER DOCUMENTS WHICH SPECIFY PROCUREMENT OF PRODUCTS FROM RACAL INSTRUMENTS INC.

REVISIONS				
ZONE	REV	DESCRIPTION	DATE	APPROVED
		RELEASED PER DRN NO.		
		REVISED PER EO NO.		
		REVISED PER EO NO.		
		REVISED PER EO NO.		

NOTES:

1. SEE WIRE LIST FOR CONNECTIONS AND CONTACT ASSIGNMENT.

△2 INK STAMP PIN 1 ON END OF CONNECTOR WHERE SHOWN.

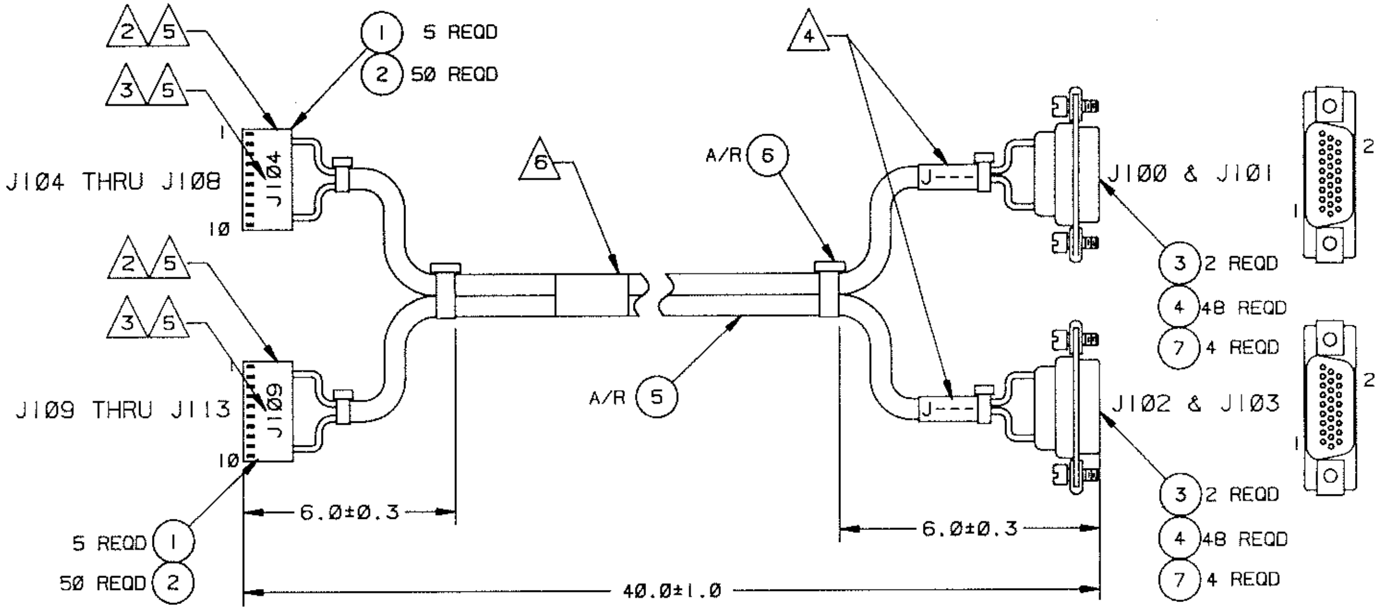
△3 WHITE INK STAMP J--- ON SIDE OF CONNECTOR APPROXIMATELY WHERE SHOWN.

△4 STAMP J--- ON APPROPRIATE SIZE SHRINK TUBING. LOCATE APPROXIMATELY WHERE SHOWN. PLACE APPROPRIATE SIZE CLEAR SHRINK TUBING OVER MARKING.

△5 SPRAY PROTECTIVE COATING (ITEM 8) AFTER MARKING.

△6 MARK 407317 ON APPROPRIATE SIZE SHRINK TUBING APPROXIMATELY WHERE SHOWN. PLACE APPROPRIATE SIZE CLEAR SHRINK TUBING OVER MARKING.

7. CONNECTORS J104-J113 WILL LATER CONNECT TO RECEIVER BLOCK, TTI P/N VGRCB-170H-R. MATING ITA CONNECTOR BLOCK IS P/N VGFCB-170.



PRECIOUS METAL INDICATOR CODE					
WEIGHTS					
STRESS					
MATERIALS & PROCESSES					
QUALITY ASSURANCE	7/20/95				FREEDOM
MANUFACTURING ENGRG T. Villa	2/17/95	PART DASH NO.	NEXT FINAL	NEXT ASSY	USED ON
ADDITIONAL APPROVALS	DATE		QTY REQ'D PER ASSY	APPLICATION	

UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES TOLERANCES ARE :

ANGLES	DECIMALS	HOLE ∅
MACH'D ± 0°30'	.XX ± .02	+.004
FORMED ± 1°0'	.XXX ± .010	-.002

DO NOT SCALE DRAWING

MATERIAL:

FINISH:

SIMILAR TO | SPECIAL HANDLING

CONTRACT NO.

APPROVALS		DATE
DRAWN	A SATTERFIELD	94-12-16
CHECKED	Ausman	95-2-1
ENGR		95-2-1
DESIGN ACTIVITY		

RACAL Instruments, Inc.
4 Goodyear St., Irvine, CA. 92718-2002

TITLE
HARNESS ASSY, 6065, TTI

SIZE	CAGE CODE	DWG NO.	REV.
B	21793	407317	A
SCALE NONE		CALC.WT ACT.WT	SHEET 1 OF 7

ENGINEERING PARTS LIST

ITEM	BIN	PART NO.	DESCRIPTION	QTY	REFERENCE
1		602193-010	CON-CAB-RCP010C.100S	10	J104-J113
2		602199-001	CONTACT, CRIMP,RECPT,28-22GA	100	W/J104-J113
3		602222	CON-D/S-PLG026C.10	4	J100-J103
4		601702-221	CONTACT,CONN,PIN	96	W/J100-J103
5		524999	WRTEF, 24GA, WHT	A/R	
6		610777	TIE-CA-LKG-.062-.075	A/R	
7		610846	LOCK SCREW ASSY	8	W/J100-J103
8		910541	POLYURETHANE CONF COAT	A/R	

DOC. NO. 407317

RACAL Instruments, Inc., 4 Goodyear St., Irvine, CA 92718

DOCUMENT TITLE	SIZE	CODE NO.	DOCUMENT NO.		REV
HARNESS ASSY, 6065, TTI	A	21793	407317		A
	DRN			SHEET 2 of 7	

ENGINEERING WIRE LIST

WIRE	FROM	TO	TYPE	PART #	WIRE LEN	REFERENCE
	BLK AAx RW 01 (J104)	Uxx-SLOT yy (J100)	CABLE	407317		SYSTEM WIRE LIST
	BLK AAx RW 02 (J105)	Uxx-SLOT yy (J100)	CABLE	407317		
	BLK AAx RW 03 (J106)	Uxx-SLOT yy (J100-J101)	CABLE	407317		
	BLK AAx RW 04 (J107)	Uxx-SLOT yy (J101)	CABLE	407317		
	BLK AAx RW 05 (J108)	Uxx-SLOT yy (J101)	CABLE	407317		
	BLK AAx RW 06 (J109)	Uxx-SLOT yy (J102)	CABLE	407317		
	BLK AAx RW 07 (J110)	Uxx-SLOT yy (J102)	CABLE	407317		
	BLK AAx RW 08 (J111)	Uxx-SLOT yy (J102-J103)	CABLE	407317		
	BLK AAx RW 09 (J112)	Uxx-SLOT yy (J103)	CABLE	407317		
	BLK AAx RW 10 (J113)	Uxx-SLOT yy (J103)	CABLE	407317		

This system wirelist serves as a template for incorporating this harness assembly into the overall system wirelist. It does not in any way affect the fabrication of this harness assembly.

DOC. NO. 407317

RACAL Instruments, Inc., 4 Goodyear St., Irvine, CA 92718

DOCUMENT TITLE	SIZE	CODE NO.	DOCUMENT NO.	REV
HARNESS ASSY, 6065, TTI	A	21793	407317	A
	DRN		SHEET 3 of 7	

ENGINEERING WIRE LIST

WIRE	FROM	TO	TYPE	PART #	WIRE LEN	REFERENCE
1	J104-1 602199-001	J100-1 601702-221	24AWG WHT	524999	40"	GND
2	J104-2 602199-001	J100-2 601702-221	24AWG WHT	524999	40"	TXD-A+
3	J104-3 602199-001	J100-3 601702-221	24AWG WHT	524999	40"	TXD-A-
4	J104-4 602199-001	J100-4 601702-221	24AWG WHT	524999	40"	RXD-A+
5	J104-5 602199-001	J100-5 601702-221	24AWG WHT	524999	40"	RXD-A-
6	J104-6 602199-001	J100-6 601702-221	24AWG WHT	524999	40"	RTS-A
7	J104-7 602199-001	J100-7 601702-221	24AWG WHT	524999	40"	CTS-A
8	J104-8 602199-001	J100-8 601702-221	24AWG WHT	524999	40"	DSR-A
9	J104-9 602199-001	J100-9 601702-221	24AWG WHT	524999	40"	DCD-A
10	J104-10 602199-001	J100-10 601702-221	24AWG WHT	524999	40"	GND
11	J105-1 602199-001	J100-11 601702-221	24AWG WHT	524999	40"	TXD-B+
12	J105-2 602199-001	J100-12 601702-221	24AWG WHT	524999	40"	TXD-B-
13	J105-3 602199-001	J100-13 601702-221	24AWG WHT	524999	40"	RXD-B+
14	J105-4 602199-001	J100-14 601702-221	24AWG WHT	524999	40"	RXD-B-
15	J105-5 602199-001	J100-15 601702-221	24AWG WHT	524999	40"	RTS-B
16	J105-6 602199-001	J100-16 601702-221	24AWG WHT	524999	40"	CTS-B
17	J105-7 602199-001	J100-17 601702-221	24AWG WHT	524999	40"	DSR-B
18	J105-8 602199-001	J100-18 601702-221	24AWG WHT	524999	40"	DCD-B
19	J105-9 602199-001	J100-19 601702-221	24AWG WHT	524999	40"	GND
20	J105-10 602199-001	J100-20 601702-221	24AWG WHT	524999	40"	+5V
21	J106-1 602199-001	J100-21 601702-221	24AWG WHT	524999	40"	+12V
22	J106-2 602199-001	J100-22 601702-221	24AWG WHT	524999	40"	-12V
23	J106-3 602199-001	J100-23 601702-221	24AWG WHT	524999	40"	GND
24	J106-4 602199-001	J100-24 601702-221	24AWG WHT	524999	40"	GND
25	J106-5 602199-001	NO CONNECT				

DOC. NO. 407317

RACAL Instruments, Inc., 4 Goodyear St., Irvine, CA 92718

DOCUMENT TITLE	SIZE	CODE NO.	DOCUMENT NO.		REV
HARNESS ASSY, 6065, TTI	**A**	**21793**	**407317**		**A**
	DRN			SHEET 4 of 7	

ENGINEERING WIRE LIST

WIRE	FROM	TO	TYPE	PART #	WIRE LEN	REFERENCE
26	J106-6 602199-001	NO CONNECT				
27	J106-7 602199-001	J101-20 601702-221	24AWG WHT	524999	40"	+5V
28	J106-8 602199-001	J101-21 601702-221	24AWG WHT	524999	40"	+12V
29	J106-9 602199-001	J101-22 601702-221	24AWG WHT	524999	40"	-12V
30	J106-10 602199-001	J101-23 601702-221	24AWG WHT	524999	40"	GND
31	J107-1 602199-001	J101-1 601702-221	24AWG WHT	524999	40"	GND
32	J107-2 602199-001	J101-2 601702-221	24AWG WHT	524999	40"	TXD-C+
33	J107-3 602199-001	J101-3 601702-221	24AWG WHT	524999	40"	TXD-C-
34	J107-4 602199-001	J101-4 601702-221	24AWG WHT	524999	40"	RXD-C+
35	J107-5 602199-001	J101-5 601702-221	24AWG WHT	524999	40"	RXD-C-
36	J107-6 602199-001	J101-6 601702-221	24AWG WHT	524999	40"	RTS-C
37	J107-7 602199-001	J101-7 601702-221	24AWG WHT	524999	40"	DTS-C
38	J107-8 602199-001	J101-8 601702-221	24AWG WHT	524999	40"	DSR-C
39	J107-9 602199-001	J101-9 601702-221	24AWG WHT	524999	40"	DCD-C
40	J107-10 602199-001	J101-10 601702-221	24AWG WHT	524999	40"	GND
41	J108-1 602199-001	J101-11 601702-221	24AWG WHT	524999	40"	TXD-D+
42	J108-2 602199-001	J101-12 601702-221	24AWG WHT	524999	40"	TXD-D-
43	J108-3 602199-001	J101-13 601702-221	24AWG WHT	524999	40"	RXD-D+
44	J108-4 602199-001	J101-14 601702-221	24AWG WHT	524999	40"	RXD-D-
45	J108-5 602199-001	J101-15 601702-221	24AWG WHT	524999	40"	RTS-D
46	J108-6 602199-001	J101-16 601702-221	24AWG WHT	524999	40"	DTS-D
47	J108-7 602199-001	J101-17 601702-221	24AWG WHT	524999	40"	DSR-D
48	J108-8 602199-001	J101-18 601702-221	24AWG WHT	524999	40"	DCD-D
49	J108-9 602199-001	J101-19 601702-221	24AWG WHT	524999	40"	GND
50	J108-10 602199-001	J101-24 601702-221	24AWG WHT	524999	40"	GND

DOC. NO. 407317

RACAL Instruments, Inc., 4 Goodyear St., Irvine, CA 92718

DOCUMENT TITLE	SIZE	CODE NO.	DOCUMENT NO.		REV
HARNESS ASSY, 6065, TTI	**A**	**21793**	**407317**		**A**
	DRN			SHEET 5 of 7	

ENGINEERING WIRE LIST

WIRE	FROM	TO	TYPE	PART #	WIRE LEN	REFERENCE
51	J109-1 602199-001	J102-1 601702-221	24AWG WHT	524999	40"	GND
52	J109-2 602199-001	J102-2 601702-221	24AWG WHT	524999	40"	TXD-E+
53	J109-3 602199-001	J102-3 601702-221	24AWG WHT	524999	40"	TXD-E-
54	J109-4 602199-001	J102-4 601702-221	24AWG WHT	524999	40"	RXD-E+
55	J109-5 602199-001	J102-5 601702-221	24AWG WHT	524999	40"	RXD-E-
56	J109-6 602199-001	J102-6 601702-221	24AWG WHT	524999	40"	RTS-E
57	J109-7 602199-001	J102-7 601702-221	24AWG WHT	524999	40"	CTS-E
58	J109-8 602199-001	J102-8 601702-221	24AWG WHT	524999	40"	DSR-E
59	J109-9 602199-001	J102-9 601702-221	24AWG WHT	524999	40"	DCD-E
60	J109-10 602199-001	J102-10 601702-221	24AWG WHT	524999	40"	GND
61	J110-1 602199-001	J102-11 601702-221	24AWG WHT	524999	40"	TXD-F+
62	J110-2 602199-001	J102-12 601702-221	24AWG WHT	524999	40"	TXD-F-
63	J110-3 602199-001	J102-13 601702-221	24AWG WHT	524999	40"	RXD-F+
64	J110-4 602199-001	J102-14 601702-221	24AWG WHT	524999	40"	RXD-F-
65	J110-5 602199-001	J102-15 601702-221	24AWG WHT	524999	40"	RTS-F
66	J110-6 602199-001	J102-16 601702-221	24AWG WHT	524999	40"	CTS-F
67	J110-7 602199-001	J102-17 601702-221	24AWG WHT	524999	40"	DSR-F
68	J110-8 602199-001	J102-18 601702-221	24AWG WHT	524999	40"	DCD-F
69	J110-9 602199-001	J102-19 601702-221	24AWG WHT	524999	40"	GND
70	J110-10 602199-001	J102-20 601702-221	24AWG WHT	524999	40"	+5V
71	J111-1 602199-001	J102-21 601702-221	24AWG WHT	524999	40"	+12V
72	J111-2 602199-001	J102-22 601702-221	24AWG WHT	524999	40"	-12V
73	J111-3 602199-001	J102-23 601702-221	24AWG WHT	524999	40"	GND
74	J111-4 602199-001	J102-24 601702-221	24AWG WHT	524999	40"	GND
75	J111-5 602199-001	NO CONNECT				

DOC. NO. 407317

RACAL Instruments, Inc., 4 Goodyear St., Irvine, CA 92718

DOCUMENT TITLE	SIZE	CODE NO.	DOCUMENT NO.		REV
HARNESS ASSY, 6065, TTI	A	21793	407317		A
	DRN			SHEET 6 of 7	

ENGINEERING WIRE LIST

WIRE	FROM	TO	TYPE	PART #	WIRE LEN	REFERENCE
76	J111-6 602199-001	NO CONNECT				
77	J111-7 602199-001	J103-20 601702-221	24AWG WHT	524999	40"	+5V
78	J111-8 602199-001	J103-21 601702-221	24AWG WHT	524999	40"	+12V
79	J111-9 602199-001	J103-22 601702-221	24AWG WHT	524999	40"	-12V
80	J111-10 602199-001	J103-23 601702-221	24AWG WHT	524999	40"	GND
81	J112-1 602199-001	J103-1 601702-221	24AWG WHT	524999	40"	GND
82	J112-2 602199-001	J103-2 601702-221	24AWG WHT	524999	40"	TXD-G+
83	J112-3 602199-001	J103-3 601702-221	24AWG WHT	524999	40"	TXD-G-
84	J112-4 602199-001	J103-4 601702-221	24AWG WHT	524999	40"	RXD-G+
85	J112-5 602199-001	J103-5 601702-221	24AWG WHT	524999	40"	RXD-G-
86	J112-6 602199-001	J103-6 601702-221	24AWG WHT	524999	40"	RTS-G
87	J112-7 602199-001	J103-7 601702-221	24AWG WHT	524999	40"	DTS-G
88	J112-8 602199-001	J103-8 601702-221	24AWG WHT	524999	40"	DSR-G
89	J112-9 602199-001	J103-9 601702-221	24AWG WHT	524999	40"	DCD-G
90	J112-10 602199-001	J103-10 601702-221	24AWG WHT	524999	40"	GND
91	J113-1 602199-001	J103-11 601702-221	24AWG WHT	524999	40"	TXD-H+
92	J113-2 602199-001	J103-12 601702-221	24AWG WHT	524999	40"	TXD-H-
93	J113-3 602199-001	J103-13 601702-221	24AWG WHT	524999	40"	RXD-H+
94	J113-4 602199-001	J103-14 601702-221	24AWG WHT	524999	40"	RXD-H-
95	J113-5 602199-001	J103-15 601702-221	24AWG WHT	524999	40"	RTS-H
96	J113-6 602199-001	J103-16 601702-221	24AWG WHT	524999	40"	DTS-H
97	J113-7 602199-001	J103-17 601702-221	24AWG WHT	524999	40"	DSR-H
98	J113-8 602199-001	J103-18 601702-221	24AWG WHT	524999	40"	DCD-H
99	J113-9 602199-001	J103-19 601702-221	24AWG WHT	524999	40"	GND
100	J113-10 602199-001	J103-24 601702-221	24AWG WHT	524999	40"	GND

DOC. NO. 407317

RACAL Instruments, Inc., 4 Goodyear St., Irvine, CA 92718

DOCUMENT TITLE	SIZE	CODE NO.	DOCUMENT NO.	REV
HARNESS ASSY, 6065, TTI	A	21793	407317	A
	DRN		SHEET 7 of 7	

APPENDIX A

SAMPLE PROGRAM

Following is a single QUICK BASIC program showing programming examples of several of the functions of the Model 6065.

```
' $INCLUDE: 'C:\GPIB-PC\QBDECL4.BAS'
REM ***************************************************************** */
REM NAME: Model 6065  SAMPLE PROGRAM            By: Racal Instruments   */
REM                                             Date: 9-28-92           */
REM FUNCTION: Familiarize User with functions                           */
REM                                                                     */
REM           Demonstrate using SCPI command set to control             */
REM                                                                     */
REM           Model 6065 VXIbus Serial Interface Module                 */
REM                                                                     */
REM Programming                                                         */
REM Environment: PC with DOS and Quick Basic                            */
REM                                                                     */
REM              National VXI GPIB-PCIIA Handler                        */
REM                                                                     */
REM                                                                     */
REM                                                                     */
REM                                                                     */
REM ***************************************************************** */
DIM resp AS STRING * 1000
DIM lin  AS STRING * 100


REM First establish your programming environment with necessary libraries
REM
REM

send1$ = "short_message"
send2$ = "#0ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890"
send3$ = "Hello, World"


REM     Second make contact with the controller

        CALL IBFIND("CTRL0", cr%)
        IF cr% < 0 THEN
           PRINT "Can't find the controller."
           STOP
        END IF

REM   Next Locate the VXI Module

        CALL IBFIND("VXITECH", bd%)
        IF bd% < 0 THEN
           CLS
           LOCATE 10, 10: PRINT "Unable to locate or communicate with the VT-1."
```

```
            LOCATE 11, 10: PRINT "Check power and setup."
            STOP
         END IF


         CALL IBWRT(bd%, "*RST")

REM      Each channel has a transmit buffer TCHx and Receive Buffer RCHx
REM      associated with the controlling hardware UART. You may alter the Baud,
REM      parity, Number of bits (5,6,7,8), pacing method of flow control and
REM      standard interface by sending the following module commands.


REM                          To select 4800 Baud for channel 3
REM
         CALL IBWRT(bd%, "SYSTEM: COMMUNICATE:SERIAL 3 :RECEIVE:BAUD 4800")
REM                                      OR ,   identically>
         CALL IBWRT(bd%, "ser3:baud 4800")
REM
         CALL IBWRT(bd%, "SYSTEM: COMMUNICATE:SERIAL 3 :TRANSMIT:BAUD 4800")
REM                                      OR ,   identically>
         CALL IBWRT(bd%, "ser3:TRAN:baud 4800")
REM
REM      Commands are not case sensitive, and only the full command word or
REM      it's specified abbreviation is accepted.
REM      Command abbreviations are usually four characters long.  See the manual
REM      for each valid shortening.
REM      Transmit autobaud is provided so the transmit baud follows the setting
REM      of the receiver.

         CALL IBWRT(bd%, "system:communicate:serial 3:transmit:auto ON")
REM                                   OR
         CALL IBWRT(bd%, "ser 3:tran:auto ON")

REM      Here  are the commands to set parity and word structure:

         CALL IBWRT(bd%, "ser 3:rec:par EVEN")

REM                                    AND
         CALL IBWRT(bd%, "syst:comm:ser3:rec:bits 7")
REM                                   OR
         CALL IBWRT(bd%, "ser3:bits 7")
REM                                    AND
         CALL IBWRT(bd%, "syst:comm:ser:sbits 2")
REM                                   OR
         CALL IBWRT(bd%, "ser3:sbit 2")

REM      Perhaps you want to select RS-422 on channel 4, then type:

         CALL IBWRT(bd%, "syst:comm:ser4:standard 422")
REM                                   OR
         CALL IBWRT(bd%, "SER4:STAN 422")

REM      To select handshaking, you have two concerns, One set the desired
REM      pacing mode, and two set valid start and stop thresholds.
REM
         CALL IBWRT(bd%, "ser3:rec:pace XON")
```

```
REM                                OR to set hardware handshaking select:

        CALL IBWRT(bd%, "ser3:control:RTS IBFull")

REM                                 AND

        CALL IBWRT(bd%, "ser3:REC:PACE:THReshold:STOP 3000")

REM                                 AND complementing

        CALL IBWRT(bd%, "ser3:REC:PACE:THR:START 500")



REM     Examples for setting and reading transmit XON/XOFF or CTS handshaking on
REM     channel 4 follow:


        CALL IBWRT(bd%, "ser4:cont:CTS ON")
REM                                OR
        CALL IBWRT(bd%, "ser4:tran:paceXON")
REM                               AND

        CALL IBWRT(bd%, "ser4:cont:CTS?")
        CALL ibrd(bd%, lin$)

REM                                   OR
        CALL IBWRT(bd%, "ser4:tran:pace?")
        CALL ibrd(bd%, lin$)

        CALL IBWRT(bd%, "")


REM     The standard Model 6065 with 128K buffer memory has default buffer
REM     allocations of 8K, which the user may reassign with the "trace:points"
REM     To reserve 4K for each transmit and receive on channel 2 type:

        CALL IBWRT(bd%, "TRACe:POINts RCH2,4096")

        CALL IBWRT(bd%, "TRACe:POINts TCH2,4096")

REM     To query the number of characters in these buffers type:

        CALL IBWRT(bd%, "TRACe:DATA:LENGth? RCH1")

        CALL ibrd(bd%, lin$)

        CALL IBWRT(bd%, "TRACe:DATA:LENGth? TCH1")

        CALL ibrd(bd%, lin$)

REM     At this time the response will be 0  .


REM     Command Verification
REM     Any of the commands can be verified by sending the system error request,
REM     "SYSTEM:ERROR?", OR "SYS:ERR?" which will respond: "No error". The Model
       6065
REM     has a two message error queue for any error output. If more than two
REM     errors occur, the second is lost and replaced with "queue overflow".
REM     Now do some house-keeping for future use.
REM     The fixed length string allocates space for the driver's output to US.


REM     Now that you have located the Model 6065 and set the standard interface,
REM     channel baud, and communications word structure, let's send a message
REM     out serial channel 2 in RS-232 Standard Interface.
REM     Select data handling mode and variable length strings for output as
REM     follows:

        CALL IBWRT(bd%, "FORMat:data 2 PACKed")

        CALL IBWRT(bd%, "TERM:LENGth 2 0")

REM     To send a brief "Hello, World" message to channel 2 send the following.

        CALL IBWRT(bd%, "trigger:AUTO 2 1")

REM     trigger:auto sets the output channel 2 to character mode.
REM     In this mode, characters are transmitted as available from the
REM     IBWRT( ) routine.


        CALL IBWRT(bd%, "trace:data tch2,#0 Hello, World")


REM     The Hello message now appears out channel 2 with the configuration:
REM     4800 baud, 7 bits, EVEN Parity, 2 stop bits.

REM     Block Mode
REM     Another transmission mode is BLOCK, which provides for building a
REM     message for one time or repetitive transmission on command trigger or
REM     timed interval without further program intervention. To demonstrate,
REM     channel 2 will continuously transmit "Start of test with any data here"
REM     plus a data pattern of any length to fit in the TCHx buffer, perhaps:
REM     "ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890@"

        CALL IBWRT(bd%, "TRIGger:AUTO 2 0")

        CALL IBWRT(bd%, "TRIGger:SEQuence:SOURce 2 TIMer")

        CALL IBWRT(bd%, "TRIGger:SEQuence:TIMer 2 1")

        CALL IBWRT(bd%, "TRAC:DATA TCH2, #0Start a new test with any data here.")

        FOR time = 1 TO 10

                CALL IBWRT(bd%, "TRAC:DATA TCH2," + send2$)
        NEXT time


REM     Now the transmit buffer TCH2 should have 36 + 10*36 = 396 characters.
REM     The number of characters in any buffer may be read with a length query.

        CALL IBWRT(bd%, "TRACe:DATA:LENGth? TCH2")

        CALL ibrd(bd%, resp$)

        PRINT "number of characters in transmit 2 "; LEFT$(resp$, ibcnt%)

REM     The command requesting the size is sent to the Model 6065 module.  Then
the
```

```
REM     response must be read. One of the Basic subroutines supplies with
REM     the National GPIB handler is IBRD called with the address of a fixed
REM     length string as shown here. Notice above "DIM resp AS STRING * 4000".


REM     Start continuous retransmission by sending a trigger. Stop the timed
REM     retransmission with "trigger:tim 2: 0".


        CALL IBWRT(bd%, "TRIGger 2")


        CALL IBWRT(bd%, "trig:seq:tim 2 0")

REM     This stops timed retransmission while maintaining the transmit buffer.

REM     Receive data is constantly put into the channel receive buffer without
REM     a command. The data must be removed from the buffer with the
REM     "TRACE:DATA? RCHx" command as shown below. Once it is read, pointer are
REM     updated and cannot be "re-read". Note resp$ is large here. Alternately,
REM     a readback length may be set then short "lines" may be read.


REM     CALL ibwrt(bd%, "TRACe:data? RCH2")

REM     CALL IBRD(bd%, resp$)

REM     To set a 36 char fixed length readback length use:

        CALL IBWRT(bd%, "term:leng 2 36")

REM     To read the buffer back in small segments use:


        CALL IBWRT(bd%, "term:leng 2 36")

        CALL IBWRT(bd%, "TRAC:DATA:LENG? rch2")
        CALL ibrd(bd%, lin$)

        IF VAL(LEFT$(lin$, ibcnt%)) > 35 THEN
           CALL IBWRT(bd%, "TRACe:data? RCH2")

           CALL ibrd(bd%, lin$)
        END IF

        FOR lineno = 1 TO 10
            CALL IBWRT(bd%, "TRAC:data? RCH2")

            CALL ibrd(bd%, lin$)

            PRINT LEFT$(lin$, ibcnt%)

        NEXT lineno

REM     NOTICE the messages are preceded by #236?
REM     The 2  is # of digits to follow for length.
REM     The 36 is # of character of data to follow.

REM     PRINT MID$(lin$,4,36)  Would print only the data, without #236.

        CALL IBWRT(bd%, "TRAC:DATA:LENG? rch2")
        CALL ibrd(bd%, lin$)
        WHILE VAL(LEFT$(lin$, ibcnt%)) > 35
           CALL IBWRT(bd%, "TRAC:data? RCH2")
           CALL ibrd(bd%, lin$)
           PRINT MID$(lin$, 5, 36)
           CALL IBWRT(bd%, "TRAC:DATA:LENG? rch2")
           CALL ibrd(bd%, lin$)
        WEND

END
```

APPENDIX B

INDEX

RACAL Instruments Inc. *4 Goodyear Street, Irvine, California 92718-2202*

REPAIR AND CALIBRATION REQUEST FORM

To allow us to better understand your repair requests, we suggest you use the following outline when calling and include a copy with your instrument to be sent to the Racal Repair Facility.

Model No.____________ Serial No.____________ Date____________

Company Name____________________ P.O.No.____________

Billing Address______________________________________

City State Zip

Shipping Address______________________________________

City State Zip

Technical Contact____________________ Phone Number (   )____________

Purchasing Contact____________________ Phone Number (   )____________

1. Describe, in detail, the problem and symptoms you are having. Please include all set up details, such as input/output levels, frequencies, waveform details, etc.

____

____

____

____

2. If problem is occurring when unit is in remote, please list the program strings used and the controller type.

____

____

____

3. Please give any additional information you feel would be beneficial in facilitating a faster repair time (i.e., modifications, etc.)

____

____

____

4. Is calibration data required? Yes No (please circle one)

Call before shipping:

Customer Service
(800) 722-3262

Ship instrument to:

Customer Service Department
Racal Instruments Inc.
4 Goodyear Street
Irvine, CA 92718

Note: We do not accept "collect" shipments.