

ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Find the *Acces 104-COM-2S* at our website: ***Click HERE***

ACCES I/O PRODUCTS, INC.
10623 Roselle Street, San Diego, CA 92121 • (858) 550-9559 • Fax (858) 550-7322
contactus@accesio.com • www.accesio.com

EIGHT- OR FOUR-PORT RS-422/485 PC/104 SERIAL COMMUNICATION BOARD

MODELS 104-COM-8S AND 104-COM-4S

USER MANUAL

File: M104-COM-8S.A1k

Notice

The information in this document is provided for reference only. ACCES does not assume any liability arising out of the application or use of the information or products described herein. This document may contain or reference information and products protected by copyrights or patents and does not convey any license under the patent rights of ACCES, nor the rights of others.

IBM PC, PC/XT, and PC/AT are registered trademarks of the International Business Machines Corporation.

Printed in USA. Copyright 2003, 2005 by ACCES I/O Products, Inc. 10623 Roselle Street, San Diego, CA 92121. All rights reserved.

WARNING!!

ALWAYS CONNECT AND DISCONNECT YOUR FIELD CABLING WITH THE COMPUTER POWER OFF. ALWAYS TURN COMPUTER POWER OFF BEFORE INSTALLING A BOARD. CONNECTING AND DISCONNECTING CABLES, OR INSTALLING BOARDS INTO A SYSTEM WITH THE COMPUTER OR FIELD POWER ON MAY CAUSE DAMAGE TO THE I/O BOARD AND WILL VOID ALL WARRANTIES, IMPLIED OR EXPRESSED.

Warranty

Prior to shipment, ACCES equipment is thoroughly inspected and tested to applicable specifications. However, should equipment failure occur, ACCES assures its customers that prompt service and support will be available. All equipment originally manufactured by ACCES which is found to be defective will be repaired or replaced subject to the following considerations.

Terms and Conditions

If a unit is suspected of failure, contact ACCES' Customer Service department. Be prepared to give the unit model number, serial number, and a description of the failure symptom(s). We may suggest some simple tests to confirm the failure. We will assign a Return Material Authorization (RMA) number which must appear on the outer label of the return package. All units/components should be properly packed for handling and returned with freight prepaid to the ACCES designated Service Center, and will be returned to the customer's/user's site freight prepaid and invoiced.

Coverage

First Three Years: Returned unit/part will be repaired and/or replaced at ACCES option with no charge for labor or parts not excluded by warranty. Warranty commences with equipment shipment.

Following Years: Throughout your equipment's lifetime, ACCES stands ready to provide on-site or in-plant service at reasonable rates similar to those of other manufacturers in the industry.

Equipment Not Manufactured by ACCES

Equipment provided but not manufactured by ACCES is warranted and will be repaired according to the terms and conditions of the respective equipment manufacturer's warranty.

General

Under this Warranty, liability of ACCES is limited to replacing, repairing or issuing credit (at ACCES discretion) for any products which are proved to be defective during the warranty period. In no case is ACCES liable for consequential or special damage arriving from use or misuse of our product. The customer is responsible for all charges caused by modifications or additions to ACCES equipment not approved in writing by ACCES or, if in ACCES opinion the equipment has been subjected to abnormal use. "Abnormal use" for purposes of this warranty is defined as any use to which the equipment is exposed other than that use specified or intended as evidenced by purchase or sales representation. Other than the above, no other warranty, expressed or implied, shall apply to any and all such equipment furnished or sold by ACCES.

TABLE OF CONTENTS

Chapter 1: FUNCTIONAL DESCRIPTION 5
Figure 1-1: BLOCK DIAGRAM 6
Chapter 2: INSTALLATION 7
Chapter 3: OPTION SELECTION 12
Figure 3-1: OPTION SELECTION MAP 14
Chapter 4: ADDRESS SELECTION 15
Table 4-1: STANDARD ADDRESS ASSIGNMENTS FOR COMPUTERS 15
Table 4-2: ADDRESS JUMPERS 16
Chapter 5: PROGRAMMING 17
Table 5-1: Control Block Register Map 17
Table 5-2: EEPROM Address Map 17
Table 5-3: BAUD RATE DIVISOR VALUES 19
Chapter 6: CONNECTOR PIN ASSIGNMENTS 22
Table 6-1: PIN CONNECTIONS 22
Chapter 7: SPECIFICATION 23
APPENDIX A 24
Table A-1: RS-422 SPECIFICATION SUMMARY 25
Figure A-1: Typical RS-485 Two-Wire Multidrop Network 26

Chapter 1: FUNCTIONAL DESCRIPTION

These Serial Interface Boards contains eight or four independent ports and provide effective RS-485 and RS-422 multipoint communication. Each channel may be configured to either mode. Jumpers on the board permit the choice of configuration, including termination, for each individual channel.

The boards are designed in the PC/104 format.

It's dimensions are approximately 3.775 inches X 3.550 inches. All signal connections are made through a 50 pin connector, mounted on edge of the board.

RS-485 BALANCED MODE OPERATION

The board supports RS-485 modes that use differential balanced drivers for increased range and noise immunity. The RS-485 specification defines a maximum of 32 devices on a single line. The number of devices served on a single line can be expanded by use of "repeaters".

The board also has the capability to add load resistors to terminate the communications lines. RS-485 communications requires that one transmitter supply a bias voltage to ensure a known "zero" state when all transmitters are off. Also, receiver inputs at each end of the network should be terminated to eliminate "ringing". These boards support biasing by default and supports termination by jumpers on the board. *If your application requires the transmitter to be un-biased, please contact the factory.*

The driver/receiver used, type 75176B, is capable of driving extremely long communication lines at high baud rates. It can drive up to ±60 mA on balanced lines and receive inputs as low as 200 mV differential signal superimposed on common mode noise of +12 V or -7 V. In case of communication conflict, the driver/receivers feature thermal shutdown.

COMM PORT COMPATIBILITY

Type 16550 UART's are used as the Asynchronous Communication Element (ACE) . These include a 16-byte transmit/receive buffer to protect against lost data in multitasking operating systems, while maintaining 100% compatibility with the original IBM serial port. However, the ports are not restricted to the standard COM port addresses.

Continuous address selection is available anywhere within the I/O address range 100 to 3F8 hex, and our FINDBASE program will scan I/O Bus memory-mapped addresses in your computer for available addresses that can be used without conflicting with other computer resources. This allows a port to be used as one of the four "standard" COM ports (COM1 to COM4), or to coexist alongside them, in any combination.

A crystal oscillator is located on the board. This oscillator permits precise selection of baud rate from 300 to 921,600 with the standard crystal oscillator. The standard crystal oscillator is used to generate two clock rates. One is the standard 1.8432 MHz clock. If higher baud rates are required, a 14.7456MHz rate can be selected by jumper.

COMMUNICATION MODES

The board supports Half-Duplex communications with a 2-wire cable connection. Half-Duplex allows traffic to travel in both directions, but only one way at a time. RS-485 communications commonly use the half-duplex mode since they share only a single pair of wires.

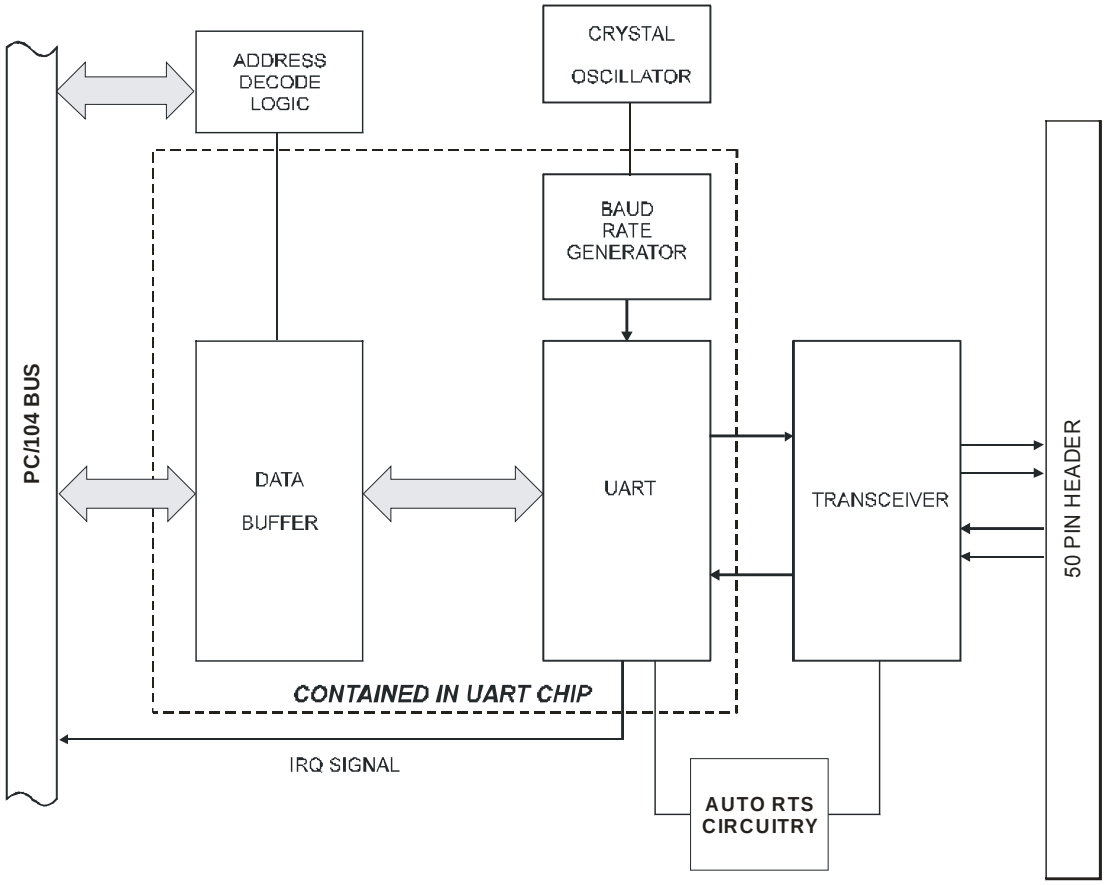
AUTO-RTS TRANSCEIVER CONTROL

In RS-485 communications, the driver must be enabled and disabled as needed, allowing all boards to share a two wire cable. The board controls the driver automatically. With automatic control, the driver is enabled when data is ready to be transmitted. The driver remains enabled for the transmission time of one character after data transfer is started and then is disabled. The receiver is disabled during RS-485 transmissions and then enabled when the transmitter driver is disabled. The board automatically adjusts it's timing to the baud rate of the data. (NOTE: Thanks to this automatic control feature, the board is ideal for use in Windows applications)

IRQ SUPPORT

The board supports the use of IRQ resources, and includes an on-board IRQ status register for use with operating systems that support this feature, such as Microsoft's Windows NT. This allows the board to use from one to five levels of IRQ to control all eight ports, greatly simplifying system configuration.

Figure 1-1: BLOCK DIAGRAM
(Only one serial channel shown)



Chapter 2: INSTALLATION

A printed Quick-Start Guide (QSG) is packed with the board for your convenience. If you've already performed the steps from the QSG, you may find this chapter to be redundant and may skip forward to begin developing your application.

The software provided with this PC/104 Board is on CD and must be installed onto your hard disk prior to use. To do this, perform the following steps as appropriate for your operating system.

CD Installation

The following instructions assume the CD-ROM drive is drive "D". Please substitute the appropriate drive letter for your system as necessary.

DOS

1. Place the CD into your CD-ROM drive.
2. Type D : Enter to change the active drive to the CD-ROM drive.
3. Type I N S T A L L Enter to run the install program.
4. Follow the on-screen prompts to install the software for this board.

WINDOWS

1. Place the CD into your CD-ROM drive.
2. The system should automatically run the install program. If the install program does not run promptly, click START | RUN and type D : I N S T A L L, click OK or press Enter.
3. Follow the on-screen prompts to install the software for this board.

LINUX

1. Please refer to linux.htm on the CD-ROM for information on installing serial ports under linux.

Installing the Hardware

Before installing the board, carefully read Chapter 3 and Chapter 4 of this manual and configure the board according to your requirements. The SETUP Program can be used to assist in configuring jumpers on the board. Be especially careful with Address Selection. If the addresses of two installed functions overlap, you will experience unpredictable computer behavior. To help avoid this problem, refer to the FINDBASE.EXE program installed from the CD. The setup program does not set the options on the board, these must be set by jumpers.

This multi-port serial communication board uses software-programmable address ranges for each UART, stored in an onboard EEPROM. Configure the address of the EEPROM using the onboard Address Selection jumper block, then use the provided Setup program to configure addresses for each onboard UART.

To Install the Board

1. Install jumpers for selected options and base address according to your application requirements, as mentioned above.
2. Remove power from the PC/104 stack.
3. Assemble standoff hardware for stacking and securing the boards.
4. Carefully plug the board onto the PC/104 connector on the CPU or onto the stack, ensuring proper alignment of the pins before completely seating the connectors together.
5. Install I/O cables onto the board's I/O connectors and proceed to secure the stack together or repeat steps 3-5 until all boards are installed using the selected mounting hardware.
6. Check that all connections in your PC/104 stack are correct and secure then power up the system.
7. Run one of the provided sample programs appropriate for your operating system that was installed from the CD to test and validate your installation.

Installing COM Ports in Windows Operating Systems

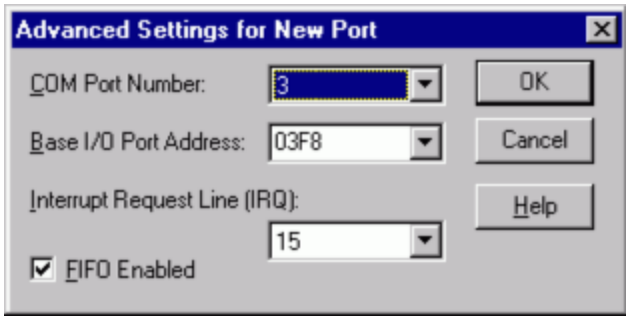
***NOTE: COM boards can be installed in virtually any operating system and we do support installation in earlier versions of windows, and are very likely to support future version as well. For use in WinCE, contact the factory for specific instructions.**

Windows NT4.0

To install the COM ports in Windows NT4 you'll need to change one entry in the registry. This entry enables IRQ sharing on multi-port COM boards. The key is **HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Serial**. The name of the value is **PermitShare** and the data should be set to **1**.

You'll then add the board's ports as COM ports, setting the base addresses and IRQs to match your board's settings.

To change the registry value, run RegEdit from the START|RUN menu option (by typing REGEDIT [ENTER] in the space provided). Navigate down the tree view on the left to find the key, and double click on the name of the value to open a dialog allowing you to set the new data value.



To add a COM port, use START|CONTROL PANEL|PORTS applet and click ADD, then enter the correct UART address and Interrupt number.

When the “Add New Port” dialog is configured click OK, but answer “Don’t Restart Now” when prompted, until you’ve added any other ports as well. Then restart the system normally, or by selecting “Restart Now.”

Windows XP

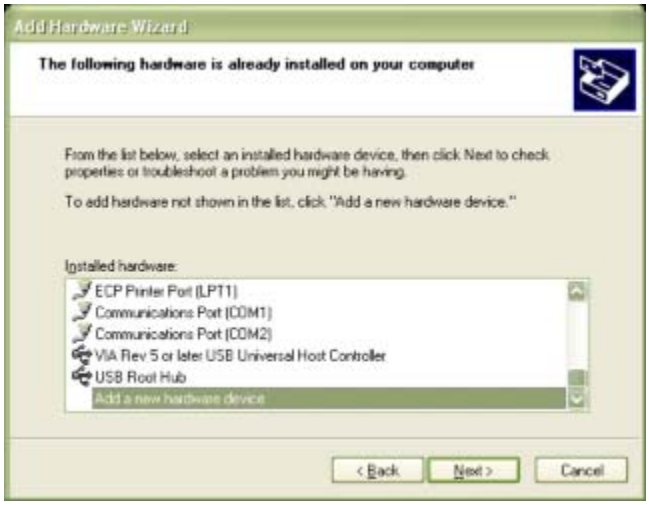
To install the COM ports in Windows XP you will be manually installing “standard” communications ports, then changing the settings for resources used by the ports to match the hardware.

Run the “Add Hardware” applet from the Control Panel.

Click “Next” at the “Welcome to the Add New Hardware Wizard” dialog.

You’ll briefly see a “...searching...” message, then

Select “Yes, I have already connected the hardware” and Click “Next”



Select “Add a new hardware device” from the bottom of the list presented and Click “Next.”

Select “Install the hardware that I manually select from a list” and Click “Next.”

Select “Ports (COM & LPT) and Click “Next”

Select “(Standard Port Types)” and “Communications Port” (the defaults), Click “Next.”

Click “Next.”



Click the “View or change resources for this hardware (Advanced)” link.

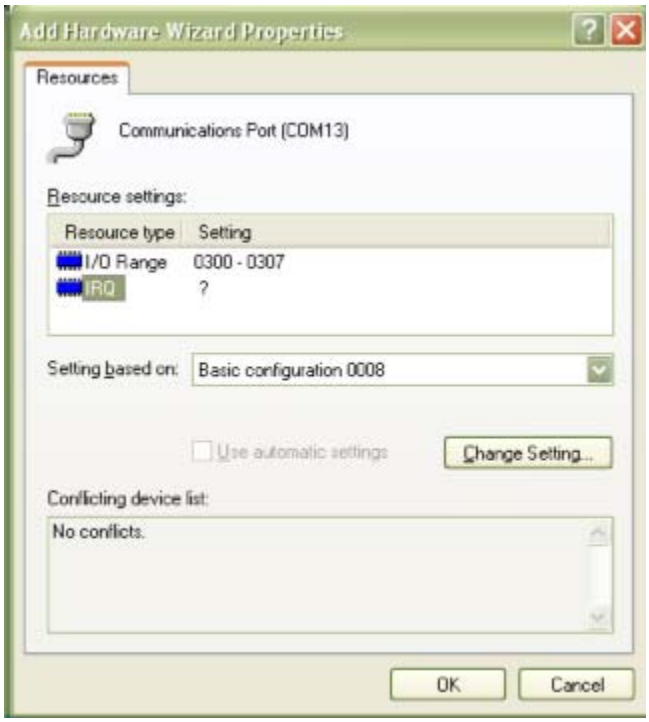


Click the “Set Configuration Manually” button.

Select “Basic Configuration 8" from the “Settings Based on:” drop-down list.

Select “I/O Range” in the “Resource Settings” box and Click the “Change Settings...” button.

Enter the base address of the board, and Click “OK”



Select “IRQ” in the “Resource Settings” box and Click the “Change Settings” button.

Enter the IRQ of the board and Click “OK”.

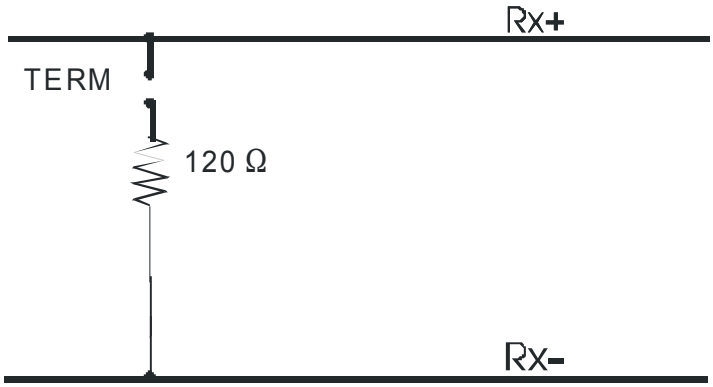
Close the “Set Configuration Manually” dialog and Click “Finish.”

Click “Do Not Reboot” if you wish to install more ports. Repeat all of the above steps, entering the same IRQ but using the configured Base address for each additional UART.

When you are done installing ports, reboot the system normally.

Chapter 3: OPTION SELECTION

To help you locate the jumpers described in this section, refer to the OPTION SELECTION MAP at the end of this section. Operation of the serial communications section is determined by jumper installation as described in the following paragraphs.



TERMINATIONS:

A transmission line should be terminated at the receiving end in its characteristic impedance. Installing a jumper at the location labeled TERM applies a 120Ω load across the transmit/receive input/output for RS-485 operation.

In RS-485 operations where there are multiple terminals, only the RS-485 ports at each end of the network should have terminating resistors as described above. Also, for RS-485 operation, there must be a bias on the RX+ and RX- lines. *If the board is not to provide that bias, contact the factory technical support.*

DATA CABLE WIRING

INTERRUPTS: The board supports IRQ 2, 3, 5, 7, 10 and 11 (unless reserved by other installed hardware). The levels are selected by writing the desired IRQ level to the appropriate address in the EEPROM and having it (them) loaded from the EEPROM into the appropriate registers. Channels A, B, C and D have individual interrupts and channels E, F, G and H share a fifth interrupt. It is necessary to load interrupt values for all of the channels. If the same interrupt is to be used for all channels, it must be entered into all five of the interrupt locations in the EEPROM.

Please note: In Windows NT, changes must be made to the System Registry to support IRQ sharing. The following is excerpted from “Controlling Multiport Serial I/O Boards” provided by Microsoft in the MSDN library.Document id: mk:@ivt:nt40res/D15/S55FC.HTM, also available in the Windows NT Resource Kit. The text enclosed in brackets (“[]”) denotes a comment.

The Microsoft serial driver can be used to control many *dumb* multiport serial boards. *Dumb* indicates that the control includes no on-board processor. Each port of a multiport board has a separate subkey under the CurrentControlSet\Services\Serial subkey in the Registry. In each of these subkeys, you must add values for **DosDevices**, **Interrupt**, **InterruptStatus**, **Port Address**, and **PortIndex** because these are not detected by the Hardware Recognizer. (For descriptions and ranges for these values, see Regentry.hlp, the Registry help file on the *Windows NT Workstation Resource Kit CD*.)

For example, if you have a board configured with the control block at address 0x300, the ports consecutive and contiguous starting at address 0x100, and an IRQ of 0x5 on all ports, the values in the Registry are:

Serial2 subkey:
PortAddress = REG_DWORD 0x100
Interrupt = REG_DWORD 5
DosDevices = REG_SZ COM3
InterruptStatus = REG_DWORD 0x500
PortIndex = REG_DWORD 1

Serial3 subkey:
PortAddress = REG_DWORD 0x108
Interrupt = REG_DWORD 5
DosDevices = REG_SZ COM4
InterruptStatus = REG_DWORD 0x500
PortIndex = REG_DWORD 2

Serial4 subkey:
PortAddress = REG_DWORD 0x110
Interrupt = REG_DWORD 5
DosDevices = REG_SZ COM5
InterruptStatus = REG_DWORD 0x500
PortIndex = REG_DWORD 3

Serial5 subkey:
PortAddress = REG_DWORD 0x118
Interrupt = REG_DWORD 5
DosDevices = REG_SZ COM6
InterruptStatus = REG_DWORD 0x500
PortIndex = REG_DWORD 4

Serial6 subkey:
PortAddress = REG_DWORD 0x120
Interrupt = REG_DWORD 5
DosDevices = REG_SZ COM7
InterruptStatus = REG_DWORD 0x500
PortIndex = REG_DWORD 5

Serial7 subkey:
PortAddress = REG_DWORD 0x128
Interrupt = REG_DWORD 5
DosDevices = REG_SZ COM8
InterruptStatus = REG_DWORD 0x500
PortIndex = REG_DWORD 6

Serial8 subkey:
PortAddress = REG_DWORD 0x130
Interrupt = REG_DWORD 5
DosDevices = REG_SZ COM9
InterruptStatus = REG_DWORD 0x500
PortIndex = REG_DWORD 7

Serial9 subkey:
PortAddress = REG_DWORD 0x138
Interrupt = REG_DWORD 5
DosDevices = REG_SZ COM10
InterruptStatus = REG_DWORD 0x500
PortIndex = REG_DWORD 8

The InterruptStatus entry being 0x500 is a bit unusual; it's the base address of the first port plus 0x400. This would normally be an alias of the first port, but the board uses this aliased address for the status register.

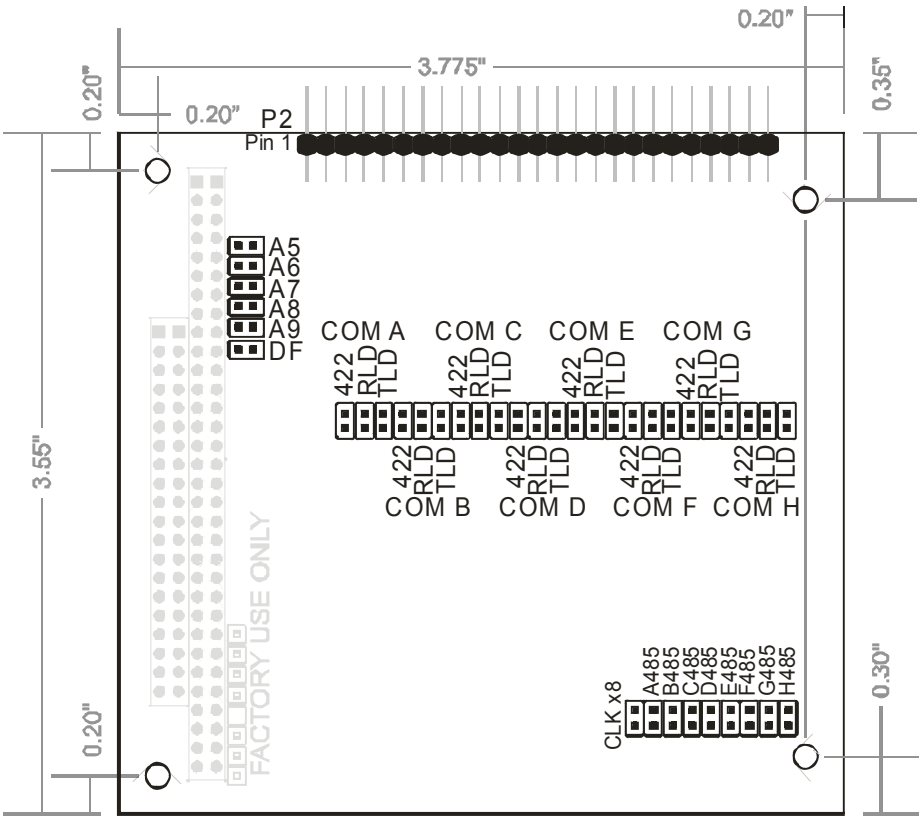


Figure 3-1: OPTION SELECTION MAP

Chapter 4: ADDRESS SELECTION

Each port's base address can be selected anywhere within an I/O address range 100-3F8 hex, providing that the address does not overlap with other functions. If in doubt, refer to the table below for a list of standard address assignments. (The primary and secondary binary synchronous communication ports are supported by the Operating System.) The FINDBASE base address locator program provided with your board will assist you to select a base address that will avoid this conflict.

Table 4-1: STANDARD ADDRESS ASSIGNMENTS FOR COMPUTERS

HEX RANGE	USAGE
000-00F	8237 DMA Controller 1
020-021	8259 Interrupt
040-043	8253 Timer
060-06F	8042 Keyboard Controller
070-07F	CMOS RAM, NMI Mask Reg, RT Clock
080-09F	DMA Page Register
0A0-0BF	8259 Slave Interrupt Controller
0C0-0DF	8237 DMA Controller 2
0F0-0F1	Math Coprocessor
0F8-0FF	Math Coprocessor
170-177	Fixed Disk Controller 2
1F0-1F8	Fixed Disk Controller 1
200-207	Game Port
238-23B	Bus Mouse
23C-23F	Alt. Bus Mouse
278-27F	Parallel Printer
2B0-2BF	EGA
2C0-2CF	EGA
2D0-2DF	EGA
2E0-2E7	GPIB (AT)
2E8-2EF	Serial Port
2F8-2FF	Serial Port
300-30F	reserved
310-31F	reserved
320-32F	Hard Disk (XT)
370-377	Floppy Controller 2
378-37F	Parallel Printer
380-38F	SDLC
3A0-3AF	SDLC
3B0-3BB	MDA
3BC-3BF	Parallel Printer
3C0-3CF	VGA EGA
3D0-3DF	CGA
3E8-3EF	Serial Port
3F0-3F7	Floppy Controller 1
3F8-3FF	Serial Port

The address jumpers determine the address of the control block; the addresses and interrupts of the ports are taken from the onboard EEPROM. The interrupt sharing register (mainly used in NT4) is referenced to the address of Channel A.

The address bytes entered into the EEPROM represent address lines A9 thru A3. The easiest way to determine the byte to write for a desired address is to divide the address by 8. For instance, a base address of 300 would be 300/8 = 60, an address of 308/8 = 61, and so on. (All addresses are in hex.)

Table 4-2: ADDRESS JUMPERS

	1st Digit		2nd Digit			
Jumper Label	A9	A8	A7	A6	A5	--
Address Line Controlled	A9	A8	A7	A6	A5	A4
Hexadecimal Value	200	100	80	40	20	10

In order to read the address jumper setup, assign a binary “1" to jumpers that are not installed and a binary “0" to jumpers installed. For example, as illustrated in the following table, jumper selection corresponds to binary 10 000x xxxx (hex 200). The “xxx” represents address lines A4, A3, A2, A1, and A0 used on the board to select individual registers, as described in the **PROGRAMMING** section of this manual.

EXAMPLE ADDRESS SETUP

Jumper Label	A9	A8	A7	A6	A5
Conversion Factors	2	1	8	4	2
Jumper Installed	NO	YES	YES	YES	YES
Binary Representation	1	0	0	0	0
Hex Representation	2		0		

Review the **ADDRESS SELECTION TABLE** carefully before selecting the board address. If the addresses of two installed functions overlap you will experience unpredictable computer behavior.

Chapter 5: PROGRAMMING

The port addresses and IRQs are selected by software through a control block; the base address of the control block is selected by jumpers. The functions within the control block are shown in the control block register map below.

Table 5-1: Control Block Register Map

Address	Read Function	Write Function
Base Address + 0	--	--
Base Address + 1	EEPROM Address	EEPROM Address
Base Address + 2	--	EEPROM Data
Base Address + 3	--	Load EEPROM To Registers

The addresses and IRQs of the ports are taken from an EEPROM on the board. In addition to automatically loading them at power-on, they can be loaded by software by a write to the control block. The addresses and interrupts are stored in the EEPROM as shown on the EEPROM address map below.

Table 5-2: EEPROM Address Map

EEPROM Address	EEPROM Data Meaning
1	Address for Channel A
2	Address for Channel B
3	Address for Channel C
4	Address for Channel D
5	Address for Channel E
6	Address for Channel F
7	Address for Channel G
8	Address for Channel H
9	IRQ for Channel A
A	IRQ for Channel B
B	IRQ for Channel C
C	IRQ for Channel D
D	IRQ for Channels E, F, G & H

As mentioned elsewhere, the addresses entered represent A3 - A9. Therefore, the data entered is the desired address, divided by 8.

When the board is first installed in a system, the ports are not necessarily at unused addresses. To prevent conflicts with other devices in the system, the board has a jumper that disables the ports, next to the base address jumpers and labeled “DF”. The control block remains enabled in this mode, allowing software to set

the port addresses appropriately. When the DF jumper is then removed, the ports will then be at the configured addresses.

To write data to the EEPROM, first write the address to the EEPROM Address register, then write to or read from the EEPROM Data register. For example, to set Channel A to address 3F8, IRQ 5, with the control block base address set to 200 (by jumpers):

Write 01 to 201.
Write 7F to 202.
Write 09 to 201.
Write 05 to 202.
Then write anything to 203 to start using these values.

All data may be entered into the EEPROM and then written to the appropriate registers with a single write to base address + 3.

SAMPLE PROGRAMS

There are two sample programs installed with the CD that is shipped with the board. These are:

Sample 1
This program is provided in C, Pascal, and QuickBASIC. It performs a test of the loopback feature of the UART. It requires no external hardware and no interrupts.

Sample 2
This program is provided in C only and demonstrates interrupt-driven RS-485 half-duplex operation. The program requires at least two computers with one board in each and a two-wire cable interconnecting them. That cable must connect the Tx pins from board 1 to the Rx pins respectively of board 2 and the Tx pins from board 2 to the Rx pins at board 1.

Board 1	to	Board 2
TRx- 3	←→	TRx- 3
TRx+ 2	←→	TRx+ 2

RS-485 PROGRAMMING

Programming the UART for RS-485 communication can be divided into three distinct sections: initialization, reception, and transmission. Initialization deals with option setup on the chip including baud rate selection. Reception deals with incoming-character processing which can be done using either polling or interrupts. Transmission deals with the process of sending the data out.

INITIALIZATION

Initializing the chip requires knowledge of the UART's register set. The first step is to set the baud rate divisor. You do this by first setting the DLAB (Divisor Latch Access Bit) high. This bit is Bit 7 at Base Address +3. In C code, the call would be:

```
outportb(BASEADDR +3,0x80);
```

You then load the divisor into Base Address +0 (low byte) and Base Address +1 (high byte). The following equation defines the relationship between baud rate and divisor:

desired baud rate = (crystal frequency) / (32 * divisor)

On the board, clock frequencies of 1.8432 MHz (Standard) and 14.7456 MHz (X8) are provided. Below is a table for the popular divisor frequencies:

Table 5-3: BAUD RATE DIVISOR VALUES

Baud Rate	Divisor (Std)	Divisor (X8)	Notes	Max Cable Length (ft)
921600	-	1		250
460800	-	2		550
230400	-	4		1400
115200	1	8		3000
57600	2	16		4000
38400	3	24		4000
28800	4	32		4000
19200	6	48		4000
14400	8	64		4000
9600	12	96	Most Common	4000
4800	24	192		4000
2400	48	384		4000
1200	96	768		4000

* Recommended maximum distances for differentially driven data cables (RS422 or RS-485) are for typical conditions. RS-232 communication lines have a maximum length of 50 feet, regardless of speed.

In C, the code to set the chip to 9600 baud is:

```
outportb(BASEADDR, 0x0C);
outportb(BASEADDR +1,0);
```

The second initializing step is to set the Line Control Register at Base Address +3. This register defines word length, stop bits, parity, and the DLAB.

Bits 0 and 1 control word length and allow word lengths from 5 to 8 bits. Bit settings are extracted by subtracting 5 from the desired word length.

Bit 2 determines the number of stop bits. There can be either one or two stop bits. If Bit 2 is set to 0, there will be one stop bit. If Bit 2 is set to 1, there will be two stop bits.

Bits 3 through 6 control parity and break enable. They are not commonly used for communications and should be set to zeroes.

Bit 7 is the DLAB discussed earlier. It must be set to zero after the divisor is loaded or else there will be no communications.

The C command to set the UART for an 8-bit word, no parity, and one stop bit is:

```
outportb(BASEADDR +3, 0x03)
```

The final initialization step is to flush the receiver buffers. You do this with two reads from the receiver buffer at Base Address +0. When done, the UART is ready to use.

RECEPTION

Reception can be handled in two ways: polling and interrupt-driven. When polling, reception is accomplished by constantly reading the Line Status Register at Base Address +5. Bit 0 of this register is set high whenever data are ready to be read from the chip. A simple polling loop must continuously check this bit and read in data as it becomes available. The following code fragment implements a polling loop and uses a value of 13, (ASCII Carriage Return) as an end-of-transmission marker:

```
do
{
        while (!(inportb(BASEADDR +5) & 1));        /*Wait until data ready*/
        data[i++]= inportb(BASEADDR);
}
while (data[i]!=13);                                /*Reads the line until null character rec'd*/
```

Interrupt-driven communications should be used whenever possible and is required for high data rates. Writing an interrupt-driven receiver is not much more complex than writing a polled receiver but care should be taken when installing or removing your interrupt handler to avoid writing the wrong interrupt, disabling the wrong interrupt, or turning interrupts off for too long a period.

The handler would first read the Interrupt Identification Register at Base Address +2. If the interrupt is for Received Data Available, the handler then reads the data. If no interrupt is pending, control exits the routine. A sample handler, written in C, is as follows:

```
readback = inportb(BASEADDR +2);
if (readback & 4)          /*Readback will be set to 4 if data are available*/
                    data[i++]=inportb(BASEADDR);
outportb(0x20,0x20);     /*Write EOI to 8259 Interrupt Controller*/
return;
```

TRANSMISSION

RS-485 transmission is simple to implement. The AUTO feature of the board automatically enables the transmitter when data are ready to send so no software enabling is required.

To transmit a string of data, the transmitter must first check Bit 5 of the Line Status Register at Base Address +5. That bit is the transmitter-holding-register-empty flag. If it is high, the transmitter has sent the data. The process of checking the bit until it goes high followed by a write is repeated until no data remains.

The following C code fragment demonstrates this process:

```
outportb(BASEADDR +4, inportb(BASEADDR +4)|0x02);
                        /*Set RTS bit without altering states of other bits*/
while(data[i]);         /*While there is data to send*/
        {
        while(!(inportb(BASEADDR +5)&0x20)); /*Wait until transmitter is empty*/
        outportb(BASEADDR,data[i]);
        i++;
        }
outportb(BASEADDR +4, inportb(BASEADDR +4)&0xFD);
                        /*Reset RTS bit without altering states of other bits*/
```

Chapter 6: CONNECTOR PIN ASSIGNMENTS

A 50-pin Male IDC Header is provided on the board. The pinout for this connector follows. Optional cabling breaks the 50 Pin Header down to 8, DB9 Male connectors.

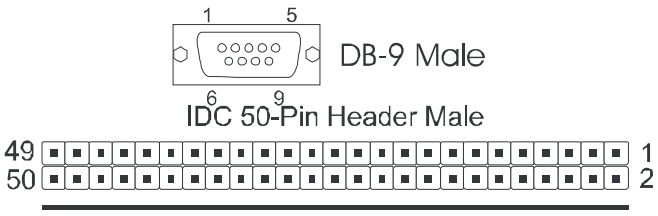


Table 6-1: PIN CONNECTIONS

Pin #	RS-485 Functions	RS-422 Functions	Pin #	RS-485 Functions	RS-422 Functions
1	Ground	Ground	26	Ch E Tx + and Rx +	Ch E Tx +
2	Ch A Tx + and Rx +	Ch A Tx +	27	Ch E Tx - and Rx -	Ch E Tx -
3	Ch A Tx - and Rx -	Ch A Tx -	28	Ground	Ground
4	Ground	Ground	29		Ch E Rx +
5		Ch A Rx +	30		Ch E Rx -
6		Ch A Rx -	31	Ground	Ground
7	Ground	Ground	32	Ch F Tx + and Rx +	Ch F Tx +
8	Ch B Tx + and Rx +	Ch B Tx +	33	Ch F Tx - and Rx -	Ch F Tx -
9	Ch B Tx - and Rx -	Ch B Tx -	34	Ground	Ground
10	Ground	Ground	35		Ch F Rx +
11		Ch B Rx +	36		Ch F Rx -
12		Ch B Rx -	37	Ground	Ground
13	Ground	Ground	38	Ch G Tx + and Rx +	Ch G Tx +
14	Ch C Tx + and Rx +	Ch C Tx +	39	Ch G Tx - and Rx -	Ch G Tx -
15	Ch C Tx - and Rx -	Ch C Tx -	40	Ground	Ground
16	Ground	Ground	41		Ch G Rx +
17		Ch C Rx +	42		Ch G Rx -
18		Ch C Rx -	43	Ground	Ground
19	Ground	Ground	44	Ch H Tx + and Rx +	Ch H Tx +
20	Ch D Tx + and Rx +	Ch D Tx +	45	Ch H Tx - and Rx -	Ch H Tx -
21	Ch D Tx - and Rx -	Ch D Tx -	46	Ground	Ground
22	Ground	Ground	47		Ch H Rx +
23		Ch D Rx +	48		Ch H Rx -
24		Ch D Rx -	49	Ground	Ground
25	Ground	Ground	50	Ground	Ground

Chapter 7: SPECIFICATION

COMMUNICATIONS INTERFACE

One 50 pin connector is provided
There are eight pins per port plus common grounds

Character Length:	5,6,7, or 8 bits
Parity:	Even, odd, or none
Stop Interval:	1, 1.5, or 2 bits
Serial Data Rates:	Up to 115.2K baud, asynchronous. A faster range of rates, up to 921.6K, is achieved by jumper selection
Multidrop:	Compatible with RS-485 specifications. Up to 32 drivers and receivers allowed on line. Driver/Receivers used are type 75ALS180

Compatible with RS-422 specifications. Up to ten receivers allowed on line.

Address:	ISA BUS address is set by jumpers on the board. Channel addresses are always loaded from non-volatile memory
Interrupts:	Individual IRQs for each channel are stored onboard in non-volatile memory
Receiver Input Sensitivity:	±200 mV differential input
Common Mode Voltage Range:	+12V to -7V Transmitter
Output Drive Capability:	60 mA with thermal shutdown.
Termination:	Jumper selectable terminations for input and output, by channel, are provided. Bias is also provided.

ENVIRONMENTAL

Operating Temperature Range:	0 to +60 °C
Storage Temperature Range:	-50 to +120 °C
Humidity:	5% to 95%, non-condensing.
Power Required:	+5 VDC at 400 mA typical, 800 mA maximum.
Size:	PC/104 format, 3.5” by 3.75”.

APPENDIX A

APPLICATION CONSIDERATIONS

INTRODUCTION

Working with RS-485 devices is not much different from working with standard RS-232 serial devices and this standard overcomes deficiencies in the RS-232 standard. First, the cable length between two RS-232 devices must be short; less than 50 feet. Second, many RS-232 errors are the result of noise induced on the cables. The RS-485 standard permits cable lengths up to 4000 feet and, because it operates in differential mode, it is more immune to induced noise.

A third deficiency of RS-232 is that more than two devices cannot share the same cable. This is also true for RS422 *but RS-485 offers all the benefits of RS422 plus allows up to 32 devices to share the same twisted pairs*. An exception to the foregoing is that multiple RS422 devices can share a single cable if only one will talk and the others will always receive.

BALANCED DIFFERENTIAL SIGNALS

The reason that RS422 and RS-485 devices can drive longer lines with more noise immunity than RS-232 devices is that a balanced differential drive method is used. In a balanced differential system, the voltage produced by the driver appears across a pair of wires. A balanced line driver will produce a differential voltage from ±2 to ±6 volts across its output terminals. A balanced line driver can also have an input “enable” signal that connects the driver to its output terminals. If the “enable” signal is OFF, the driver is disconnected from the transmission line. This disconnected or disabled condition is usually referred to as the “tristate” condition and represents a high impedance. RS-485 drivers must have this control capability. RS422 drivers may have this control but it is not always required.

A balanced differential line receiver senses the voltage state of the transmission line across the two signal input lines. If the differential input voltage is greater than +200 mV, the receiver will provide a specific logic state on its output. If the differential voltage input is less than -200 mV, the receiver will provide the opposite logic state on its output. The maximum operating voltage range is from +6V to -6V allowing for voltage attenuation that can occur on long transmission cables.

A maximum common mode voltage rating of ±7V provides good noise immunity from voltages induced on the twisted pair lines. The signal ground line connection is necessary in order to keep the common mode voltage within that range. The circuit may operate without the ground connection but may not be reliable.

Table A-1: RS-422 SPECIFICATION SUMMARY

Parameter	Conditions	Min.	Max.
Driver Output Voltage (unloaded)		4V -4V	6V -6V
Driver Output Voltage (loaded)	TERM jumpers in	2V -2V	
Driver Output Resistance			50Ω
Driver Output Short-Circuit Current			±150 mA
Driver Output Rise Time			10% unit interval
Receiver Sensitivity			±200 mV
Receiver Common Mode Voltage Range			±7V
Receiver Input Resistance			4KΩ

To prevent signal reflections in the cable and to improve noise rejection in both the RS422 and RS-485 mode, the receiver end of the cable should be terminated with a resistance equal to the characteristic impedance of the cable. (The exception is when the line is driven by an RS422 driver that is never “tristated” or disconnected from the line. In this case, the driver provides a low internal impedance that terminates the line at that end.)

NOTE

You do not have to add a terminator resistor to your cables when you use the board. Termination resistors for the RX^+ and RX^- lines are provided on the board and are placed in the circuit when you install the LOAD (LD) jumpers. (See the **Option Selection** section of this manual.)

RS-485 DATA TRANSMISSION

The RS-485 Standard allows a balanced transmission line to be shared in a party-line mode. As many as 32 driver/receiver pairs can share a two-wire party line network. Many characteristics of the drivers and receivers are the same as in the RS422 Standard. One difference is that the common mode voltage limit is extended and is +12V to -7V. Since any driver can be disconnected (or tristated) from the line, it must withstand this common mode voltage range while in the tristate condition.

RS-485 Two-Wire Multidrop Network

The following illustration shows a typical multidrop or party line network. Note that the transmission line is terminated on both ends of the line but not at drop points in the middle of the line.

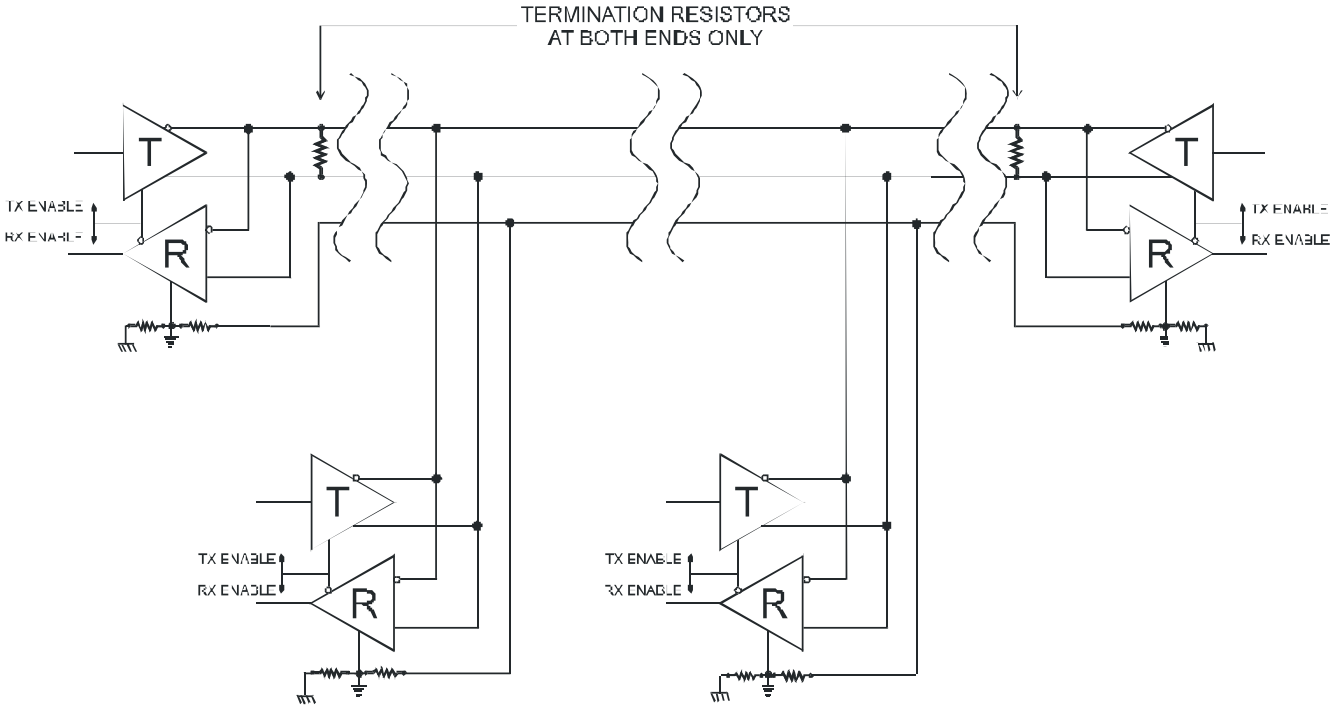


Figure A-1: Typical RS-485 Two-Wire Multidrop Network

Customer Comments

If you experience any problems with this manual or just want to give us some feedback, please email us at: ***manuals@accesio.com.*** Please detail any errors you find and include your mailing address so that we can send you any manual updates.

ACCES I/O PRODUCTS, INC.
10623 Roselle Street, San Diego CA 92121
Tel. (858)550-9559 FAX (858)550-7322
www.accesio.com

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP