ACCES I/O PRODUCTS INC
10623 Roselle Street, San Diego, CA 92121
TEL (858)550-9559   FAX (858)550-7322

MODEL COM422

USER MANUAL

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.

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

List of Figures

List of Tables

Chapter 1: Introduction

The COM422 card was designed for industrial applications, and is economical to use. It is half size and can be installed in either short or long slots of IBM PC/XT/AT or compatible computers. The card can be used for either RS422 serial communications or RS485 communications. (The RS485 specification allows multiple transmitters and receivers to communicate over a two-wire "party line" bus.) Type NS16550 UARTs are used as the Asynchronous Communication Element (ACE). Use of the same ACE as in IBM original equipment makes the card 100% compatible with existing programs when the card address is set as either COM 1 or COM 2.

However, use of the COM422 card is not restricted to COM 1 or COM 2. Continuous address selection is available anywhere within the I/O address range 000 to 3FF hex.

A crystal oscillator is located on the card. This oscillator permits precise selection of baud rate from 50 to 56000.

The output transceiver used, the new generation type 75176, is capable of driving extremely long communication lines at high baud rates. It is capable of driving up to 60mA on balanced lines and receiving inputs as low as 200mV differential signal superimposed on common mode noise of maximum -7V/+12V. In case of communication conflict, the transceivers feature thermal shutdown.

For increased noise immunity, the communication lines are loaded at the receiver and biased at the transmitter. If you have a card model COM422(T), then optional transzorbs have been added to the I/O lines to further increase immunity to high noise environment.

Two LED indicators are provided on the COM422 card. The LEDs blink to indicate activity on the transmitting and receiving lines and are useful for problem diagnosis.

In addition to dual, differential Transmit and Receive lines, single-ended buffered RTS and CTS lines are provided on the output connector. The RTS line can be used to control the Transmitter and Receiver. The CTS line is normally controlled by software. If you are not controlling CTS under software, CTS must be pulled up to +5VDC for the card to operate. This is easily done by installing a jumper at a jumper position labeled CTS. If you prefer, you can omit that jumper and install a jumper in the mating connector between pins 5 & 9. When that jumper is installed, it serves to check for proper installation of the communication cable. To check for proper cable connection, you can read the CTS bit by software.

Full-Duplex, Half-Duplex, or Simplex configuration can be selected by jumper options.

Specification

- Multipoint: Compatible with RS422 and RS485 specifications. Up to 32 drivers and receivers allowed on line. Serial communications ACE used is type NS16550. Transceivers used are type 75176.
- Input Common Mode Voltage: -7V to +12V.
- Receiver Input Sensitivity: +/- 200mV, differential input. Thermal shutdown protection.
- Output Drive Capability: +/- 60 mA max., differential. Thermal shutdown protection.
- Tranzorbs: Optional Tranzorbs can be added to I/O lines for additional protection (Option T).
- RTS and CTS: Lines available on output connector.
- Auxiliary Power Output: +5VDC at 20mA at connector pin 9.
- Baud Rate: 50 to 56000 baud. Crystal oscillator provided.
- Address: Eight bytes, continuously mappable within I/O address range 000 to 3FF hex.

Environmental

- Operating Temperature Range: 0 °C. to +60 °C.
- Storage Temperature Range: -50 °C. to +120 °C.
- Humidity: 5% to 95%, non-condensing.
- Power Required: +5VDC at 400 mA typical.
- Size: 4.9" long. Can mount in half or full-size slots.
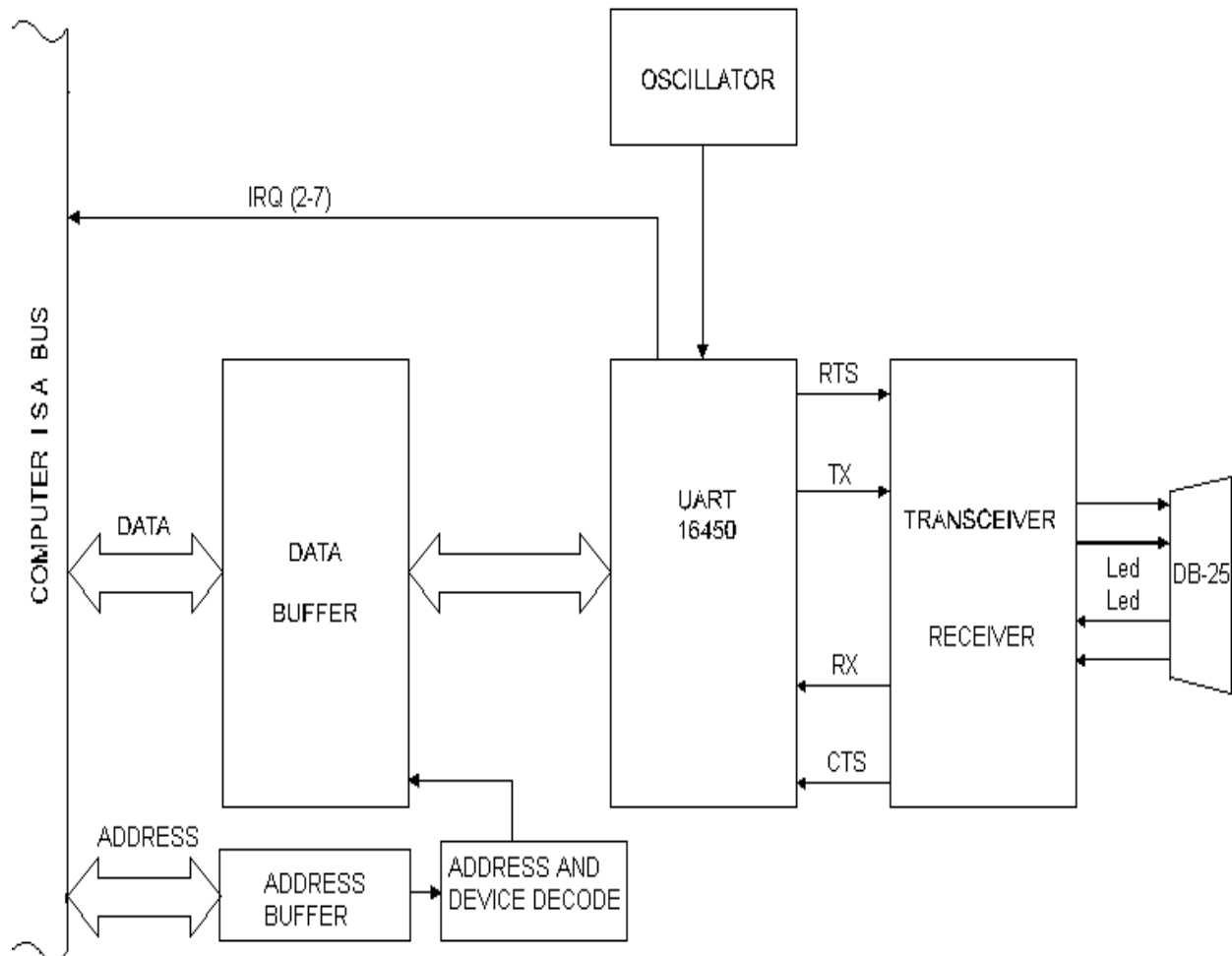
Figure 1-1: COM422 Block Diagram

Chapter 2: Installation

The software provided with this card is contained on either one CD or multiple diskettes and must be installed onto your hard disk prior to use. To do this, perform the following steps as appropriate for your software format and operating system. Substitute the appropriate drive letter for your CD-ROM or disk drive where you see d: or a: respectively in the examples below.

CD Installation

DOS/WIN3.x
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 card.

WIN95/98/NT
a. Place the CD into your CD-ROM drive.
b. The CD should automatically run the install program after 30 seconds. If the install program does not run, click START | RUN and type d:install, click OK or press `Enter`.
c. Follow the on-screen prompts to install the software for this card.

3.5-Inch Diskette Installation

As with any software package, you should make backup copies for everyday use and store your original master diskettes in a safe location. The easiest way to make a backup copy is to use the DOS DISKCOPY utility.

In a single-drive system, the command is:

`D` `I` `S` `K` `C` `O` `P` `Y`   `A` `:`   `A` `:` `Enter`

You will need to swap disks as requested by the system.
In a two-disk system, the command is:

`D` `I` `S` `K` `C` `O` `P` `Y`   `A` `:`   `B` `:` `Enter`

This will copy the contents of the master disk in drive A to the backup disk in drive B.

To copy the files on the master diskette to your hard disk, perform the following steps.
1.　　Place the master diskette into a floppy drive.
2.　　Change the active drive to the drive that has the diskette installed. For example, if the diskette is in drive A, type A : Enter.
3.　　Type I N S T A L L Enter and follow the on-screen prompts.

Directories Created on the Hard Disk

The installation process will create several directories on your hard disk. If you accept the installation defaults, the following structure will exist.

[CARDNAME]
Root or base directory containing the SETUP.EXE setup program used to help you configure jumpers and calibrate the card.

DOS\PSAMPLES:	A subdirectory of [CARDNAME] that contains Pascal samples.
DOS\CSAMPLES:	A subdirectory of [CARDNAME] that contains "C" samples.
Win32\language:	Subdirectories containing samples for Win95/98 and NT.

WinRisc.exe
A Windows dumb-terminal type communication program designed for RS422/485 operation. Used primarily with Remote Data Acquisition Pods and our RS422/485 serial communication product line. Can be used to say hello to an installed modem.

ACCES32
This directory contains the Windows 95/98/NT driver used to provide access to the hardware registers when writing 32-bit Windows software. Several samples are provided in a variety of languages to demonstrate how to use this driver. The DLL provides four functions (InPortB, OutPortB, InPort, and OutPort) to access the hardware.

This directory also contains the device driver for Windows NT, ACCESNT.SYS. This device driver provides register-level hardware access in Windows NT. Two methods of using the driver are available, through ACCES32.DLL (recommended) and through the DeviceIOControl handles provided by ACCESNT.SYS (slightly faster).

SAMPLES

Samples for using ACCES32.DLL are provided in this directory. Using this DLL not only makes the hardware programming easier (MUCH easier), but also one source file can be used for both Windows 95/98 and WindowsNT. One executable can run under both operating systems and still have full access to the hardware registers. The DLL is used exactly like any other DLL, so it is compatible with any language capable of using 32-bit DLLs. Consult the manuals provided with your language's compiler for information on using DLLs in your specific environment.

VBACCES

This directory contains sixteen-bit DLL drivers for use with VisualBASIC 3.0 and Windows 3.1 only. These drivers provide four functions, similar to the ACCES32.DLL. However, this DLL is only compatible with 16-bit executables. Migration from 16-bit to 32-bit is simplified because of the similarity between VBACCES and ACCES32.

PCI

This directory contains PCI-bus specific programs and information. If you are not using a PCI card, this directory will not be installed.

SOURCE

A utility program is provided with source code you can use to determine allocated resources at run-time from your own programs in DOS.

PCIFind.exe

A utility for DOS and Windows to determine what base addresses and IRQs are allocated to installed PCI cards. This program runs two versions, depending on the operating system. Windows 95/98/NT displays a GUI interface, and modifies the registry. When run from DOS or Windows3.x, a text interface is used. For information about the format of the registry key, consult the card-specific samples provided with the hardware. In Windows NT, NTioPCI.SYS runs each time the computer is booted, thereby refreshing the registry as PCI hardware is added or removed. In Windows 95/98/NT PCIFind.EXE places itself in the boot-sequence of the OS to refresh the registry on each power-up.

This program also provides some COM configuration when used with PCI COM ports. Specifically, it will configure compatible COM cards for IRQ sharing and multiple port issues.

WIN32IRQ

This directory provides a generic interface for IRQ handling in Windows 95/98/NT. Source code is provided for the driver, greatly simplifying the creation of custom drivers for specific needs. Samples are provided to demonstrate the use of the generic driver. Note that the use of IRQs in near-real-time data acquisition programs requires multi-threaded application programming techniques and must be considered an intermediate to advanced programming topic. Delphi, C++ Builder, and Visual C++ samples are provided.

Findbase.exe

DOS utility to determine an available base address for ISA bus , non-Plug-n-Play cards. Run this program once, before the hardware is installed in the computer, to determine an available address to give the card. Once the address has been determined, run the setup program provided with the hardware to see instructions on setting the address switch and various option selections.

Poly.exe

A generic utility to convert a table of data into an nth order polynomial. Useful for calculating linearization polynomial coefficients for thermocouples and other non-linear sensors.

Risc.bat

A batch file demonstrating the command line parameters of RISCTerm.exe.

RISCTerm.exe

A dumb-terminal type communication program designed for RS422/485 operation. Used primarily with Remote Data Acquisition Pods and our RS422/485 serial communication product line. Can be used to say hello to an installed modem. RISCTerm stands for Really Incredibly Simple Communications TERMinal.

Installing the Card

The COM422 card can be installed in either a long or short slot of an IBM PC/XT/AT or compatible computer. Before installing the card, carefully read Chapter 3, Option Selection and Chapter 4, Address Selection and configure the card according to your requirements. Use the FINDBASE program provided on the CD with the card to select the base address for the card. Use the program called SETUP to select other options.

Be especially careful with address selection. If the addresses of two installed functions overlap, you will experience unpredictable computer behavior.

To Install the Card
1. Turn off computer power.
2. Remove the computer cover.
3. Remove the blank mounting bracket.
4. Install jumpers for selected options. See Chapter 3.
5. Select the base address on the card. See Chapter 4.
6. Install the card in an I/O expansion slot. Important: Make sure that the card mounting bracket is properly screwed into place and that there is a positive chassis ground.
7. Install the I/O cable.
8. Inspect for proper fit of the card and cable and tighten screws.
9. Turn the computer ON and observe the LED indicators. The LED's will blink when there is any activity on the communication lines.
10. Turn the computer OFF and replace the computer cover.

To ensure that there is minimum susceptibility to EMI and minimum radiation, it is important that there be a positive chassis ground. Also, proper EMI cabling techniques (cable connect to chassis ground at the I/O connector, twisted-pair wiring, and, in extreme cases, ferrite level of EMI protection) must be used for input/output wiring. CE-marked versions of COM422 meet the requirements if EN50081-1:1992 (Emissions), EN50082-1:1992, and EN60950:1992 (Safety).

Chapter 3: Option Selection

Refer to Figure 3-1, COM422 Option Selection Map when reading this section of the manual. Card operation is determined by jumper installation as described in the following paragraphs.

CTS

Serial communications will not operate without this signal. The function of the CTS jumper is to provide the signal when it is not externally supplied. If you prefer, you cn omit this jumper and install a jumper between CTS (pin 5) and +5 VDC (pin 9) on the mating D connector P1. This jumper in the mating connector provides a handy diagnostic tool because the card will not operate unless the mating connector is properly installed.

Slot 8

In order to use the COM422 Serial Interface card in a short slot, install a jumper at this position.

Terminations and Bias

A transmission line should be terminated at the receiving end in its characteristic impedance.   Installing a jumper at the location labeled LD applies a 136Ω load across the input for RS422 mode and across the transmit/receive input/output for RS485 operation. When noise is a potential problem on long lines, the terminating resistor should be divided and its center point grounded to help reduce noise voltage pickup.   To accomplish this, also install a jumper at the position marked LD GND for 68Ω termination resistance on the positive and negative branches of the receiving line.

In RS485 operations, where there are multiple terminals, only the RS485 ports at each end of the network should have terminating resistors as described above. If the card is to have an ungrounded load, do as above except do not install the LDGND jumper. (See Appendix A, Application Considerations.) Also, for RS485 operation, there must be a bias on the RX+ and RX- lines. If the COM422 card is to provide that bias, install jumpers at the locations labeled +BIAS and -BIAS.

RTS Control

for RS485 operation, installing a jumper at the location labeled RTS allows the state of the RTS line to be controlled by the UART. Without this jumper, the RTS signal will always be high and allow the port to start a transmission at any time.
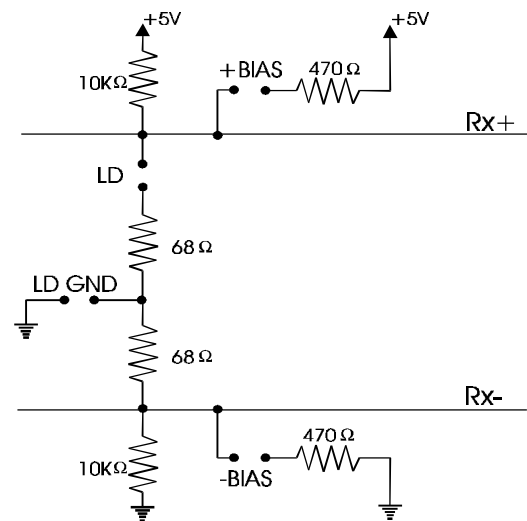
Simplex or Duplex

The receiver can be set in either Simplex or Duplex by installing jumpers marked SX or DX. Simplex mode is intended for one-way communication; either transmit or receive. Duplex mode allows transmission or reception either simultaneous or alternatively. In Duplex mode the receiver is always enabled and the echo of the port's transmission is fed back to the UART receiver. In Simplex mode the receiver is always enabled.

Connections for Simplex (transmit only or receive only) are:

Receive Only: DX and FDX jumpers, connector pins 12 and 13
Transmit Only: SX and FDX, connector pins 24 and 25

Full or Half-Duplex

Either Full or Half-Duplex can be selected by installing jumpers at the locations labeled FDX or HDX respectively. Full-Duplex allows simultaneous bi-directional communications and is selected by installing the FDX jumper. Half-Duplex allows bi-directional transmitter and receiver operation but only one at a time. Proper selection depends on the wire connections used to connect the two serial ports.

RS422 operation requires a jumper at FDX and RS485 operation requires a jumper at HDX.

Connections for Half-Duplex (transmit and receive taking turns) and Full-Duplex (transmit and receive at same time) are as follows:

Half-Duplex, 2-wire, w/echo: DX, HDX, and RTS jumpers, connector pins 12 and 13.
Half-Duplex, 2-wire, no echo: SX, HDX, and RTS jumpers, connector pins 12 and 13.
Half-Duplex, 2-wire, w/echo: DX, FDX, RTS jumpers, TX on connector pins 12 and 24, RX on connector pins 13 and 25.
Full-Duplex, 4-wire, no echo: DX and FDX, TX on connector pins 24 and 25, RX on connector pins 12 and 13.

Interrupts

Interrupt levels IRQ2 through IRQ7 are available. Select the desired level by installing a jumper in one of the locations marked IN2 through IN7.
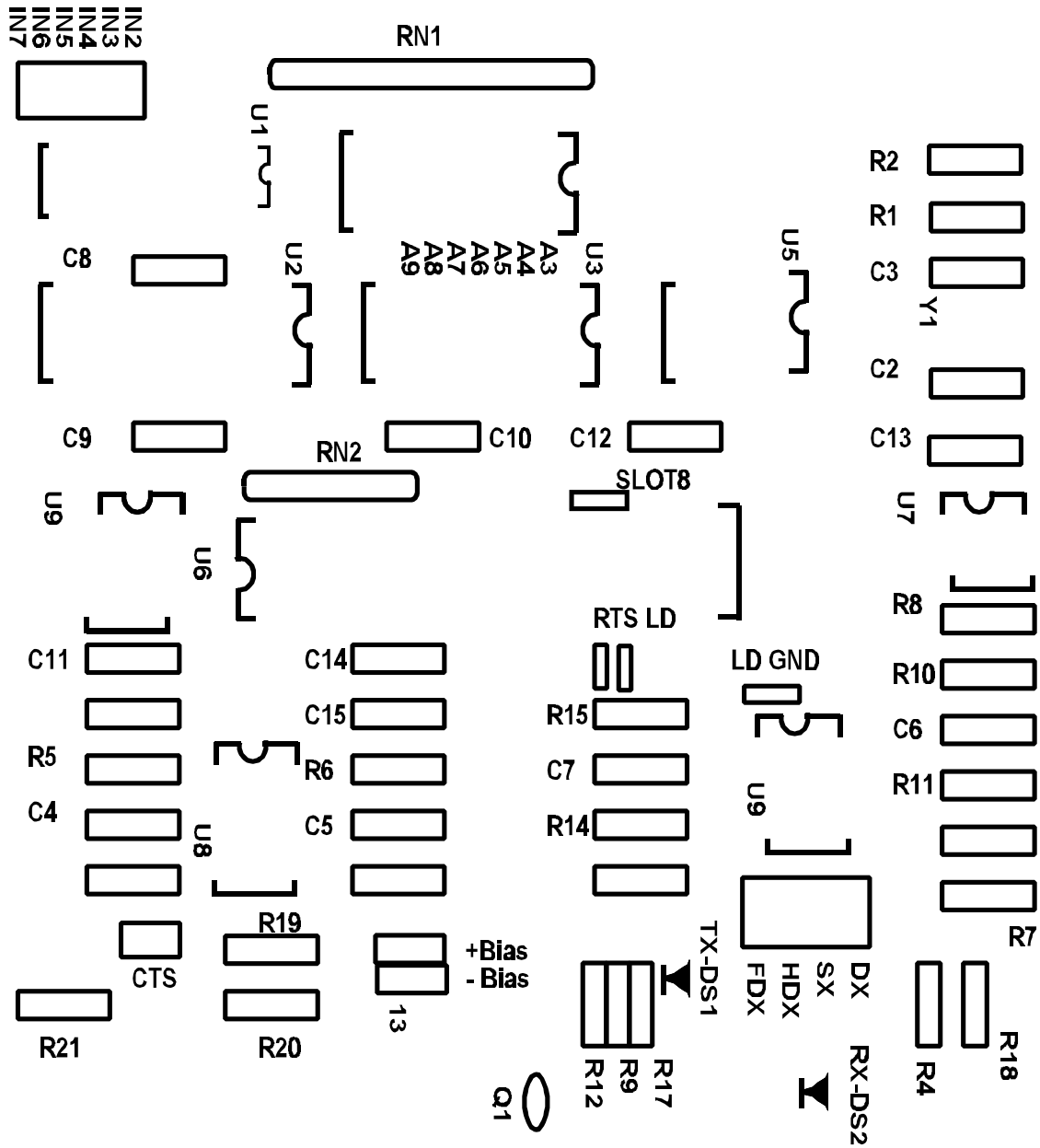
Figure 3-1: COM422 Option Selection Map

Chapter 4: Address Selection

The COM422 Serial Interface card base address can be selected anywhere within an I/O address range 100-3FF hex in AT's (except 1F0 through 1F8) and 200-3FF in XT's, providing that the address does not overlap with other functions. If in doubt, refer to the tables below for a list of standard address assignments. (Note: The primary and secondary binary synchronous communication ports are supported by the Operating System.) The base address locator program FINDBASE provided will assist you to select a base address that will avoid any problem.

Hex Range	Usage
000-00F	DMA Chip 8237A-5
020-021	Interrupt 8259A
040-043	Timer 8253-5
060-063	PPI 8255A-5
080-083	DMA Page Register
0AX	NMI Mask Register
0CX	Reserved
0EX	Reserved
100-1FF	Not Usable
200-20F	Game Control
210-217	Expansion Unit
220-24F	Reserved
278-27F	Reserved
2F0-2F7	Reserved
2F8-2FF	Asynchronous Comm'n (Secondary)
300-31F	Prototype Card
320-32F	Fixed Disk
378-37F	Printer
380-38C**	SDLC Communications
380-389**	Binary Synchronous Comm'n (Secondary)
3A0-3A9	Binary Synchronous Comm'n (Primary)
3B0-3BF	IBM Monochrome Display/Printer
3C0-3CF	Reserved
3D0-3DF	Color/Graphics
3E0-3E7	Reserved
3F0-3F7	Diskette
3F8-3FF	Asynchronous Comm'n (Primary)

**These options cannot be used together-addresses overlap.

Table 4-1: Standard Address Assignments for PC and PC/XT Computers

Hex Range	Usage
000-01F	DMA Controller 1
020-03F	INT Controller 1, Master
040-05F	Timer
060-06F	8042 (Keyboard)
070-07F	Real Time Clock, NMI Mask
080-09F	DMA Page Register
0A0-0BF	INT Controller 2
0C0-0DF	DMA Controller 2
0F0	Clear Math Coprocessor Busy
0F1	Reset Coprocessor
0F8-0FF	Arithmetic Processor
1F0-1F8	Fixed Disk
200-207	Game I/O
278-27F	Parallel Printer Port 2
2F8-2FF	Asynchronous Comm'n (Secondary)
300-31F	Prototype Card
360-36F	Reserved
378-37F	Parallel Printer Port 1
380-38F	SDLC or Binary Synchronous Comm'n 2
3A0-3AF	Binary Synchronous Comm'n 1
3B0-3BF	Monochrome Display/Printer
3C0-3CE	Local Area Network
3D0-3DF	Color/Graphic Monitor
3F0-3F7	Floppy Diskette Controller
3F8-3FF	Asynchronous Comm'n (Primary)

Table 4-2: Standard Address Assignments for 286/386/486 Computers

Address Setup switches are marked A3-A9. In order to configure the desired address, assign "0" to all Address Setup switches turned OFF and assign "1" to all Address Setup switches turned ON. These 1's and 0's are a binary representation of the base address. This binary number is then converted to a hexadecimal number.

Switch Label	A9	A8	A7	A6	A5	A4	A3
Addr. Line Contr'lld	A9	A8	A7	A6	A5	A4	A3

For example, as illustrated in the following table, switch selection corresponds to binary 10 1101 1xxx (or hex 2D8). The "xxx" represents address lines A2, A1, and A0 used on the card to select individual registers. See Chapter 5, Programming of this manual.

Switch Label	A9	A8	A7	A6	A5	A4	A3
Setup	ON	OFF	ON	ON	OFF	ON	ON
Binary Represent'n	1	0	1	1	0	1	1
Conversion Factor	2	1	8	4	2	1	8
Hex Represent'n	2		D				8

Review the Address Selection Table carefully before selecting the card address. If the addresses of two installed functions overlap you will experience unpredictable computer behavior.

Chapter 5: Programming

Sample Programs

There are sample programs provided with the COM422 card in C, Pascal, QuickBASIC, and several Windows languages. DOS samples are located in the DOS directory and Windows samples are located in the WIN32 directory.

Windows Programming

The COM422 card installs into Windows as COM ports. Thus the Windows standard API functions can be used. In particular:
▸ CreateFile() and CloseHandle() for opening and closing a port.
▸ SetupComm(), SetCommTimeouts(), GetCommState(), and SetCommState() to set and change a port's settings.
▸ ReadFile() and WriteFile() for accessing a port.
See the documentation for your chosen language for details.

Under DOS, the process is very different. The remainder of this chapter describes DOS programming.

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 located at Base Address +3, Bit 7. In C, the call would look like:

    outportb(BASEADDR +3,0x80);

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

    desired baud rate = (crystal frequency) / (16 * divisor)

On the COM422 card, the crystal frequency is 1.8432 MHz. Commonly used divisors are: 12 for 9600 baud, 48 for 2400 baud, and 96 for 1200 baud. In C, the code to set the chip to 9600 baud is:

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

The second step in initializing the UART 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 0's.

Bit 7 is the DLAB. It must be set to 0 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 third step of the initialization sequence is to set the MODEM control register at Base Address +4. This register controls functions on some cards. Bit 1 is the Request to Send (RTS) control bit. This bit should be left low until transmission time. Bits 2 and 3 are user-designated outputs. Bit 2 may be ignored on this card. Bit 3 is used to enable interrupts and should be set high if an interrupt-driven receiver is to be used.

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 this is 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 available to be read from the chip. A simple polling loop must continuously check this bit and read in data when it becomes available. Polling is not effective at very high data rates because the program cannot do anything else when it is polling or data could be missed. The following is a code fragment that implements a polling loop and uses a value of zero as an end-of-transmission marker:

```
do
{
  while (!(inportb(BASEADDR +5) & 1));    /*Wait until data ready*/
  data[i++]=inportb(BASEADDR);
}
while (data[i]!=0);    /*Reads the line until null character rec'd*/
```

Interrupt-driven communications should be used whenever possible and is required for data speeds. Writing an interrupt-driven receiver is not much more complex than writing a polled receiver. However, care should be taken when installing or removing your interrupt handler because there is a danger of writing the wrong interrupt, or disabling the wrong interrupt, or even 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. As sample handler, written in C, is as follows:

```
do
{
 readback = inportb(BASEADDR +2);
 if(readback & 4) //*Readback will be set to 4 if data are  available*/
 data[i++]=inportb(BASEADDR);
}
while(readback!=1);
outportb(0x20,0x20);     /*Write EOI to 8259 Interrupt Controller*/
return;
```

The preceding examples accept only a single character input. The code would need to be modified to handle larger data structures.

Transmission

RS485 transmission is simple to implement. First, the RTS line should be set high by writing a 1 to Bit 1 of the modem control register at Base Address +4. The RTS line is used to toggle the transceiver from receive mode to transmit mode and vice versa. It is not carried out on the line in RS485 and not used for handshaking.

Similarly, the CTS line is not used in RS485 and should always be enabled by installing a jumper as described in the Option Selection section of this manual.

After the above are done, the card is ready to send data. 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. After all data has been transmitted, the RTS bit should be reset by writing a 0 to Bit 1 of the modem control register .

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)                                        /*While there is data to send*/
{
 while(!(inportb(BASEADDR +5)&0x20));               /*Wait until transmitter is
                                                    empty*/
 outportb(BASEADDR,*data);
 data++;
}
```

```
outportb(BASEADDR +4, inportb(BASEADDR +4)&0xFD;
                                /*Reset RTS bit without altering states of other bits*/
```

Chapter 6: Connector Pin Assignments

The popular 25-pin D subminiature connector is used for interfacing to communication lines. The connector is equipped with 4-40 threaded standoffs (female screw lock) to provide strain relief. The mating connector is AMP type 747304-2 or equivalent. We recommend using vinyl-jacketed, multiple twisted-pair cable.

Pin No.	Assignment
4	Request to Send (RTS)
5	Clear to Send (CTS)
7	Signal Ground
9	+5 V Supply
12	Receive Line + (RX+)
13	Receive Line - (RX-)
24	Transmit Line + (TX+)
25	Transmit Line - (TX-)

Table 6-1: Connector Pin Assignments

Note

For Simplex, Half-Duplex, and Full-Duplex operation, see Chapter 3, Option Selection for pin connection information.

Customer Comments

If you experience any problems with this manual or just want to give us some feedback, please email us at: *manuals@accesioproducts.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. (619)550-9559   FAX (619)550-7322
www.accesioproducts.com