

Copyright Notice

This documentation and the software included with this product are copyrighted 1998 by Advantech Co., Ltd. All rights are reserved. Advantech Co., Ltd. reserves the right to make improvements to the products described in this manual at any time without notice.

No part of this manual or software may be reproduced, copied, translated or transmitted, in any form or by any means without the prior written permission of Advantech Co., Ltd. Information provided in this manual is intended to be accurate and reliable. However, Advantech Co., Ltd. assumes no responsibility for its use, nor for any infringements of rights of third parties which may result from its use.

Acknowledgments

PC-LabCard is a trademark of Advantech Co., Ltd. IBM, PC and PC/XT/AT are trademarks of International Business Machines Corporation. MS-DOS, MASM, QuickBASIC, Microsoft C and MS-PASCAL are trademarks of Microsoft Corporation. Intel is a trademark of Intel Corporation. Turbo C and Turbo PASCAL are trademarks of Borland International. MOXA is a trademark of Moxa Technologies Inc.

Contents

Chapter 1: Introduction	1
Description	2
Features	3
Specifications	3
Applications	4
Chapter 2: Hardware Configuration	5
Initial Inspection	6
Jumper and Switch Locations	7
Jumper settings	8
Card Configuration	8
Card installation	13
Chapter 3: Software Installation:	15
Installation	16
Card setup	16
DOS Driver Installation	20
Driver Removal	20
Programming example	21
Chapter 4: Wiring	23
Pin assignments	24
RS-422 signal wiring	24
RS-485 signal wiring	26
Appendix A: Register structure and format	29
Register Structure	30
Programming example	33
Appendix B: PC I/O address reference	37

CHAPTER

1

Introduction

Description

The PCL-743 and PCL-745 series cards each provide two RS-422/485 serial ports. Each port has a 16C550 UART with an on-chip 16-byte FIFO buffer for reliable high-speed serial I/O. The UART buffers data into 16-byte packets before putting it on the bus. This drastically reduces the CPU's load and avoids data loss when the system is busy and cannot process the interrupt quickly, which is especially useful for high speed serial I/O under Windows.

The PCL-745B/745S cards differ from the PCL-743B/743S cards in that they utilize on-board optical isolators to protect your PC and equipment against damage from ground loops, increasing system reliability in harsh environments. In order to meet the demand for reliability, the PCL-743S/745S introduces surge protection technology. This surge protection can prevent damage from abrupt high voltage (2000 V_{DC}) sources such as lightning power line strikes.

These cards include the PC-ComLIB communication software package. This package consists of a configurable DOS driver, a programming library that supports most common languages (including C, Pascal, Visual Basic, assembly and Clipper), and DataScope. DataScope is a data viewer and terminal emulator with self-diagnostics utilities for easy troubleshooting and debugging.

An RS-485 network with automatic data flow control using RS-232 software

In RS-485 mode the cards automatically sense the direction of incoming data and switch transmission direction accordingly. This feature means your network looks and acts just like an RS-232 network. Application software written for half duplex RS-232 can be used without modification. Moreover, you can simply and quickly build an RS-485 network with just two wires.

Features

- Two independent RS-422/RS-485 serial ports
- Provides 500 V_{DC} isolation (PCL-745B/745S only)
- Provides 2000 V_{DC} surge protection (PCL-743S/745S only)
- 16C550 UARTs with on-chip 16-byte FIFO
- Transmission speeds up to 115 Kbps
- I/O address and interrupt selectable
- Wide IRQ selection: 3, 4, 5, 6, 7, 9, 10, 11, 12 or 15
- Supports 2-wire or 4-wire operation
- Supports Tx, Rx, RTS and CTS signals
- Automatic RS-485 data flow control
- Space reserved for terminator resistors
- Space reserved for optional surge protection on data lines
- Supports standard DOS COM1, COM2, COM3 and COM4 ports
- Includes PC-ComLIB Driver

Specifications

- **Ports:** 2
- **UART:** 2 x 16C550 with 16 byte FIFO
- **Signals supported:**
RS-422: TxD+, TxD-, RxD+, RxD-, RTS+, RTS-, CTS+, CTS-, and GND
RS-485: DATA+, DATA-, and GND
- **I/O addresses:** From 200H to 3FFH
- **IRQ:** 3, 4, 5, 6, 7, 9, 10, 11, 12 or 15
- **Surge protection:** 2000 V_{DC} (PCL-743S/745S only)
- **Isolation Voltage:** 500 V_{DC} (PCL-745B/745S only)
- **Power consumption:** +5 V @ 400 mA typical, 950 mA max.

- **Connectors:** Dual DB-9 male connectors
- **Operating temperature:** 32 to 140° F (0 to 60° C) (refer to IEC 68-12-1.2.3)
- **Dimensions:** 7.3" x 3.9" (185 mm x 100 mm)
- **Storage temperature:** -13 ~ 176° F (-25 ~ 80° C) (only for card)
- **MTBF:**
PCL-745B/745S: over 122,264 hrs at 25° C, Ground-fix environment
PCL-743B/743S: over 251,319 hrs at 25° C, Ground-fix environment

Applications

- PLC monitoring and control
- Serial communication interface for harsh environments
- Data entry terminal
- Remote data acquisition and control systems
- Instrument controller and distributed control systems

CHAPTER
2

**Hardware
Configuration**

Initial Inspection

You should find the following items inside the shipping package (in addition to this manual):

- PCL-743B or 745B, or 743S, or 745S interface card
- PC-ComLIB diskette
- PC-ComLIB user's manual

We carefully inspected the PCL-743/745 series mechanically and electrically before we shipped it. It should be free of marks and scratches and in perfect working order on receipt.

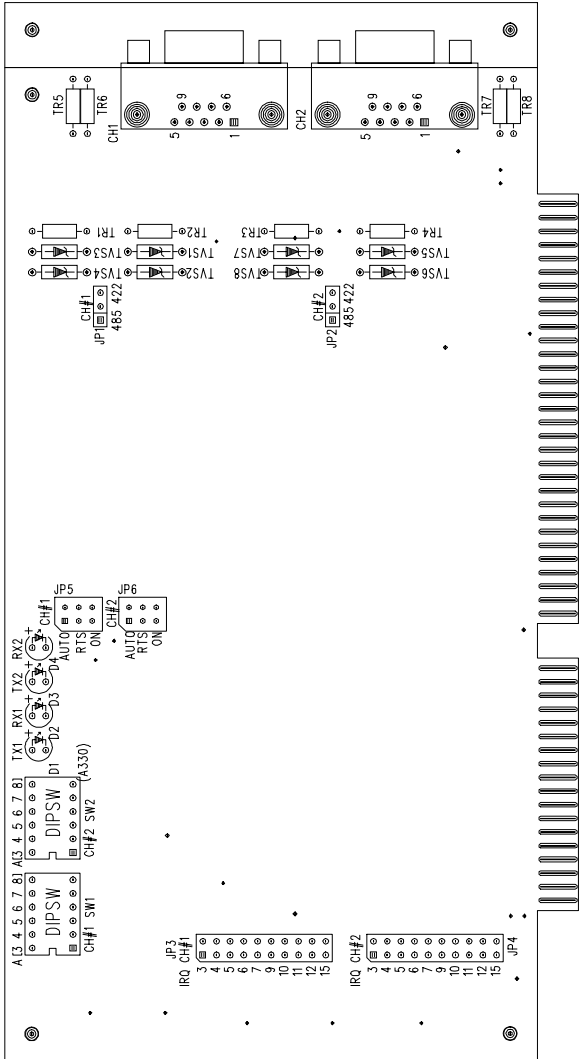
As you unpack the PCL-743/745 series, check it for signs of shipping damage (damaged box, scratches, dents, etc.). If it is damaged or it fails to meet specifications, notify our service department or your local sales representative immediately. Also notify the carrier. Retain the shipping carton and packing material for inspection by the carrier. After inspection we will make arrangements to repair or replace the unit.

When you handle the PCL-743/745 series, remove it from its protective packaging by grasping the rear metal panel. Keep the anti-vibration packing. Whenever you remove the card from the PC, store it in this package for protection.

Warning! *Discharge your body's static electric charge by touching the back of the grounded chassis of the system unit (metal) before handling the board. You should avoid contact with materials that hold a static charge such as plastic, vinyl and styrofoam. Handle the board only by its edges to avoid static damage to its integrated circuits. Avoid touching the exposed circuit connectors. We also recommend that you use a grounded wrist strap and place the card on a static dissipative mat whenever you work with it.*



Jumper and Switch Locations



Jumper settings

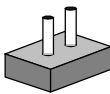
This section tells how to set the jumpers to configure your card. It gives the card default configuration and your options for each jumper.

How to set jumpers

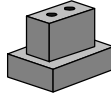
You configure your card to match the needs of your application by setting jumpers. A jumper is the simplest kind of electric switch. It consists of two metal pins and a small metal clip (often protected by a plastic cover) that slides over the pins to connect them. To “close” a jumper you connect the pins with the clip. To “open” a jumper you remove the clip. Sometimes a jumper will have three pins, labeled 1, 2 and 3. In this case you would connect either pins 1 and 2 or 2 and 3.

You may find pair of needle-nose pliers useful for setting the jumpers.

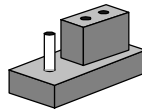
If you have any doubts about the best hardware configuration for your application, contact your local distributor or sales representative before you make any changes.



Open



Closed



Closed 2-3

Card Configuration

You can configure each of the card's ports separately. Each port has two jumpers and one six-position DIP switch. The jumpers set the port's protocol (RS-422 or RS-485) and IRQ. The DIP switch sets the port base I/O address. The following chart shows the function of each jumper and switch:

Switch and jumper functions

RS-422 or RS-485

JP1Port 1

JP2Port 2

Port IRQ

JP3Port 1

JP4Port 2

Port I/O address

SW1 Port 1

SW2 Port 2

See page 7 for jumper and DIP switch locations

Default Settings

When the board comes from the factory, both ports are set for RS-422. Port 1 is set for DOS COM1 (IRQ = 4, base I/O address = 3F8), and Port 2 is set for DOS COM2 (IRQ = 3, base I/O address = 2F8).

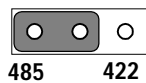
If you need to change these settings, see the following sections.

Otherwise, you can simply install the card. Note that you will need to disable your CPU card's on-board COM ports if any, or set them to alternate addresses / IRQs.

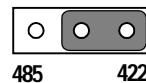
RS-422/485 selection (JP1 and JP2)

You can set each port individually for either RS-422 (the default) or RS-485 operation. Jumper JP1 configures Port 1, and JP2 configures Port 2. The figure below shows the jumper settings. See the figure on page 7 for help locating the jumpers.

RS-422/485



RS-422 (default)



I/O base address (SW1 and SW2)

You set the port base addresses using two six-position DIP switches, one for each port. Switch SW1 sets Port 1, and SW2 sets Port 2. You can choose base addresses from hex 200 to 3FF. The default settings are COM1 (hex 3F8) for Port 1, and COM2 (hex 2F8) for Port 2, as shown below:



SW1: CH#1
Base address: 3F8



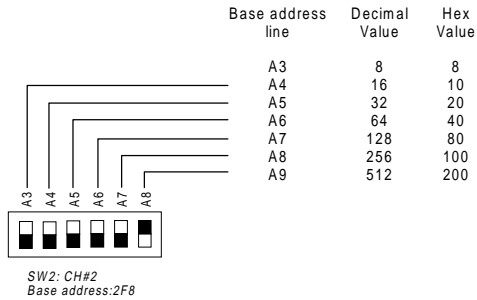
SW2: CH#2
Base address: 2F8

The following table shows the switch settings for various base addresses:

Port base address (SW1 or SW2)						
Base address	A3	A4	A5	A6	A7	A8
200-207	●	●	●	●	●	●
208-20F	○	●	●	●	●	●
2E8-2EF	○	●	○	○	○	●
*2F8-2EF (COM2)	○	○	○	○	○	●
3E8-3EF	○	●	○	○	○	○
*3F8-3FF (COM1)	○	○	○	○	○	○

○ = Off ● = On * = default

The following example shows how to set the base address to 2F8, the default setting for COM2: The switch sum is set to 2F8 = 200 + 80 + 40 + 20 + 10 + 8, (HEX).

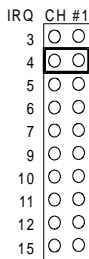


NOTE: On the PCL-743/745 series, address line A9 does not appear on the DIP switch, as it is permanently hard-wired to hex 200.

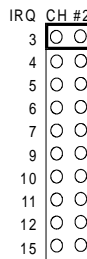
Interrupt level (IRQ) setting (JP3 and JP4)

Jumpers JP3 and JP4 set the interrupt for each port. You can choose any IRQ from 3 to 15 except for 6, 8, 13, and 14. JP3 sets Port 1 and JP4 sets Port 2. If you want to use the card as COM1 and COM2, you must set Port 1 for IRQ4 and Port 2 for IRQ3 (the default settings). If you choose different IRQ's, make sure they are not used by other cards in the system. The following figures show the card default settings:

JP3: Port 1
IRQ4 (default)



JP4: Port 2
IRQ3 (default)



Enable mode selection

Jumpers JP5 and JP6 set the driver enable mode selections of ports 1 and 2 respectively. If jumpers JP5 or JP6 are set at "AUTO," the driver automatically senses the direction of the data flow and switches the direction of transmission. No handshaking is necessary.

If jumpers JP5 or JP6 are set to RTS (request to send), a high RTS signal can enable the driver. Otherwise, the driver output remains in high impedance status. If jumpers JP5 or JP6 are set to "On," the driver is always enabled, and always in high or low status. The user must select a mode before beginning RS-422 applications.

Terminator resistor setup

You can install terminator resistors if necessary to match impedance. Each signal line (RTS, CTS) has a separate resistor. Resistors TR5 (for CTS) and TR6 (for RTS) are for port 1. Resistors TR7 (for CTS) and TR8 (for RTS) are for port 2.

Card installation

Warning! *Turn off your PC's power supply whenever you install or remove the PCL-743/745 or its cables. Static electricity can easily damage computer equipment. Ground yourself by touching the chassis of the computer (metal) before you touch any boards. See the static warning on page 6*



1. Turn off the computer. Turn the power off to any peripheral devices (such as printers and monitors).
2. Disconnect the power cord and any other cables from the back of the computer. Turn the PC if necessary to gain access to the cables.
3. Remove the PC's cover (refer to your user's guide if necessary).
4. Locate the expansion slots or passive backplane (at the rear of the PC) and choose any unused slot.
5. Remove the screw that secures the expansion slot cover to the PC (save the screw to secure the interface card retaining bracket). Remove the anti-vibration card clamp if supplied.
6. Carefully grasp the upper edge of the PCL-743/745 card. Align the hole in the retaining bracket with the hole on top of the expansion slot. Align the gold striped edge connector with the expansion slot socket. Press the board firmly into the socket.
7. Replace the screw in the expansion slot retaining bracket. Replace anti-vibration card holder.
8. Replace the PC's cover. Connect the cables you removed in step 2. Turn the computer power on.

The board is now installed in the computer. See Chapter 4 for information on cabling.

CHAPTER
3

**Software
Installation:**

Installation

Make a duplicate copy of the driver diskette in case the original disk becomes lost or damaged. Copy the files to a sub-directory on your hard disk if you desire.

The PCL-743/745 comes with the PC-ComLIB software package. PC-ComLIB provides software drivers for DOS which supports most common languages, including C, PASCAL, Visual Basic, Quick Basic, assembly and Clipper. The DOS driver will automatically use four bytes of the 16C550 UART's FIFO buffer. PC-ComLIB also includes the DataScope data viewer, terminal emulator and self-diagnostics utilities for easy troubleshooting and debugging. Please see the PC-ComLIB manual for more detailed information.

Card setup

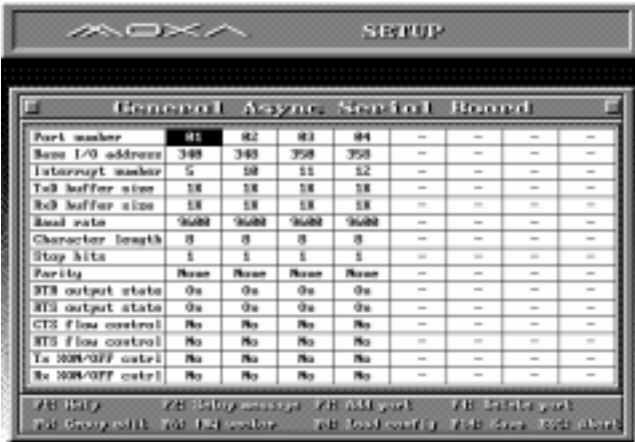
The PCL-743/745 driver determines the configuration of the installed cards by reading a data file, `GEN-DRV.CNF`. When you first install the PCL-743/745, and each time you change the jumper settings, you will need to run the card setup program to save the settings to the configuration file.

To invoke the setup program insert the driver disk in your computer and select the appropriate drive (or change to the directory on your hard disk containing the program files). Type `SETUP` from the DOS prompt and press `ENTER`. You will then see the following screen:



Driver selection screen

After the screen shows up, move the cursor bar (using the arrow keys or the mouse) to the General Serial I/O field and press ENTER. Then press the PgDn key to get to the PCL-743/745 series setup screen, as shown on the following page.

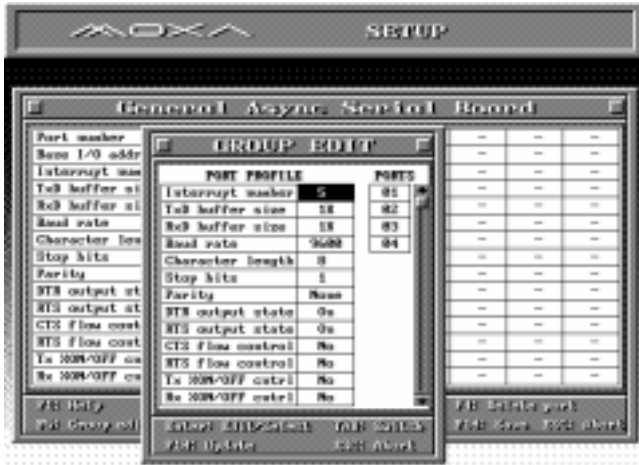


Port setup

Here you can set the configuration for each of the PCL-743/745 series cards installed in the system. Set the IRQ, I/O address and port number to match the card's DIP switch settings. Do this by moving the highlight to the field you want to set up. Press ENTER and a menu will appear, allowing you to select the correct setting.

The setup program also controls the port number assignments for each card. When you use a driver function in your program, you will identify the ports by these assignments. After you have set the IRQ, base address and port assignments, press F10 to save the settings or ESC to return to the previous page.

Note: *Make sure that the base address and IRQ selected do not conflict with any other cards you may have installed in your system.*



Group Edit menu

If you wish to define I/O controls such as the baud rate and buffer size for all installed ports, you can use the PORT SETUP screen. If any of your ports have the same setup, you can define them all at one time by pressing F5 to bring up the GROUP EDIT screen. When you are finished setting up the ports, press the ESC key to return to the previous screen. Press F10 to save the new configuration or ESC to quit without saving. The setup program will then create a new configuration data file GEN-DRV . CNF .

DOS Driver Installation

You must install the PCL-743/745 card driver (GEN-DRV . EXE) before you run any application programs, including the DataScope utilities. If you are installing the driver for the first time, you will need to run the setup program (described in the previous section) to save the initial status of the PCL-743/745.

Because the driver is a TSR (terminate and stay resident) program, you can execute it at any time. You do not have to install it in the CONFIG.SYS file.

To install the driver, change to the directory or floppy disk containing the driver files and type GEN-DRV. You should soon see the following message:

```
PC-ComLIB communication Driver (Ver x.xx)
```

```
General Serial I/O Card : aa Port
```

```
DEVICE DRIVER SETUP O.K.
```

Where "aa" is the number of ports in your computer system

If the driver cannot detect any card or the GEN-DRV configuration file is not in the same directory as the driver, the following message will be displayed:

```
PC-ComLIB Communication Driver (Ver x.xx)
```

```
General Serial I/O Card Not Found
```

```
Device Driver SETUP Fail!
```

Driver Removal

To remove the driver TSR program, just type GEN-DRV/q from the DOS prompt. The following message will appear:

```
PC-ComLIB Communication Driver (Ver x.xx) Release O.K.
```

Programming example

The following code extract illustrates how the library functions can simplify card programming, reduce setup time and avoid any bugs. The following C program shows a simple data transfer test. It uses a PC-ComLIB function call to send data between Port 1 and Port 2 at a rate of 57.6 Kbps

```
/* Execute the PC-ComLIB TSR driver first*/

#include<head-c.h>
main()
{
    /* Setup PORT1 and PORT2 : bps = 57600;
       *data bits = 8; stop bits = 1; no parity */
    sio_ioctl(1,B57600,BIT_8|P_NONE|STOP_1);
    sio_ioctl(2,B57600,BIT_8|P_NONE|STOP_1);

    /* Enable communication ports */
    sio_open(1);
    sio_open(2);

    /* Transmit data on Port 1 */
    sio_write(1," Hello ",7);
    delay(200);

    /* Receive data on Port 2,store in buf1*/
    sio_linput(2,buf1,7,13);

    /* Print received data */
    printf("%s\n",buf1);

    /* Disable communication ports */
    sio_close(1);
    sio_close(2);
}
```

See the PC-ComLIB user's manual for information on programming and linking your application programs with the driver libraries. See chapter 4 for wiring information.

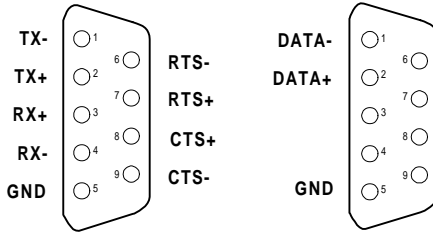
CHAPTER

4

Wiring

Pin assignments

The following figures show the pin assignments for the card's DB-9 connectors in RS-422 and RS-485 modes.



RS-422 signal wiring

The RS-422 interface wiring is based on one-to-one principles. The transmit lines on one side connect to the receive lines on the other side, and vice versa. With RS-422, you can transmit and receive data simultaneously (full duplex). The connections are as follows:

PCL-743/745 DTE (Male DB-9)		Terminal DTE
Pin	Signal	Signal
1	TxD-	RxD-
2	TxD+	RxD+
3	RxD+	TxD+
4	RxD-	TxD-
5	GND	GND
6	RTS-	CTS-
7	RTS+	CTS+
8	CTS+	RTS+
9	CTS-	RTS-

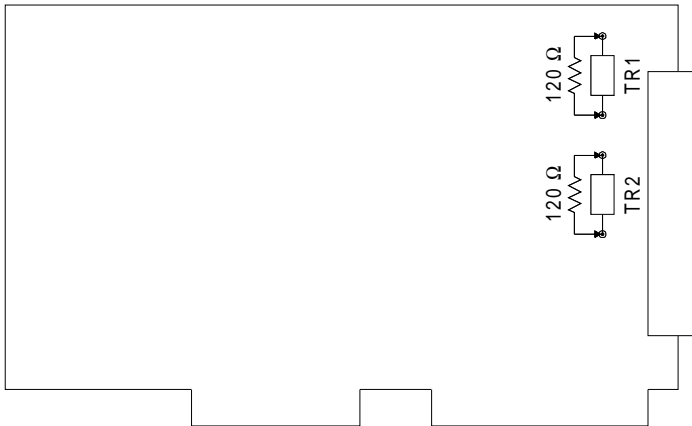
Terminator Resistors setup

The signals DSR, DTR and DCD are shorted internally on the PCL-743/745 cards when operating in RS-422 mode.

A user can solder in termination resistors if necessary for impedance matching. The card has two mounting spaces for termination resistors, but no resistors are installed at the factory. Each pair of signal lines has a separate resistor (Rx D_{\pm} -, Tx D_{\pm} -). Resistors TR1 and TR2 are for Port 1. Resistors TR3 and TR4 are for Port 2.

Resistors are usually installed at both ends of the communication lines. The value of each resistor should equal the characteristic impedance of the signal wires used (approximately 120 Ohms).

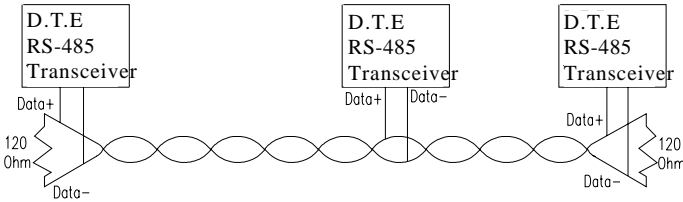
The following figure shows the resistor placement for Port 1:



RS-422 termination resistor installation

RS-485 signal wiring

The RS-485 standard supports half-duplex communication. This means that just two wires are needed to both transmit and receive data. Handshaking signals (such as RTS, Request To Send) are normally used to control the direction of the data flow and to switch the transmission accordingly. In RS-485 mode, the PCL-743/745 automatically senses the direction of the data flow and switches the transmission direction — no handshaking is necessary. This means a user can build an RS-485 network with just two wires. This RS-485 control is completely transparent to the user. The Software written for half duplex RS-232 works without the need for any modification.



RS-485 wiring topology

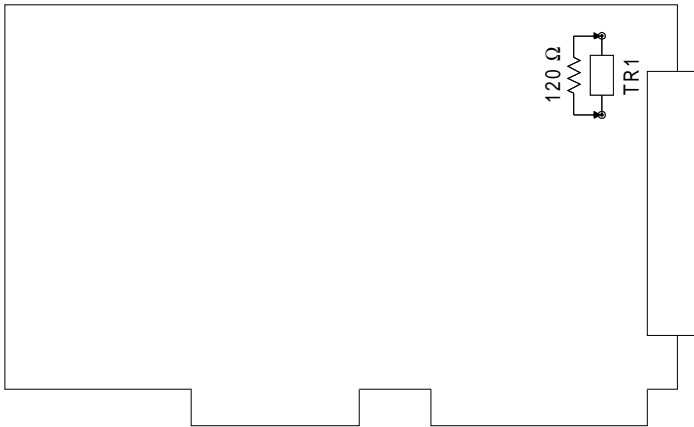
Wiring connections are as follows:

PCL-743/745 DTE (male DB-9)		Terminal DTE
Pin	Signal	Signal
2	DATA+	DATA +
1	DATA-	DATA -

Termination resistor setup

You can install termination resistors if necessary for impedance matching. The card has mounting spaces for termination resistors, but no resistors are installed at the factory. Depending on your application you may need to solder in a single resistor to handle the DATA+/DATA- pair (and a corresponding resistor on the other end of the connection). The value of the resistor should equal the characteristic impedance of the signal wires (approximately 120 Ohms).

The figure below shows the resistor placement for Port 1 (TR1). Termination resistor TR3 handles Port 2.



RS-485 Termination resistor installation

APPENDIX
A

**Register structure
and format**

Register Structure

This appendix gives short descriptions of each of the module's registers. For more information please refer to the data book for the STARTECH 16C550 UART chip.

All registers are one byte. Bit 0 is the least significant bit, and bit 7 is the most significant bit. The address of each register is specified as an offset from the port base address (BASE), selected with DIP switch SW1 or SW2.

DLAB is the "Divisor Latch Access Bit", bit 7 of BASE+3.

BASE+0 Receiver buffer register when DLAB=0 and the operation is a read.

BASE+0 Transmitter holding register when DLAB=0 and the operation is a write.

BASE+0 Divisor latch bits 0 - 7 when DLAB=1.

BASE+1 Divisor latch bits 8 - 15 when DLAB=1

The two bytes BASE+0 and BASE+1 together form a 16-bit number, the divisor, which determines the baud rate. Set the divisor as follows:

Baud rate	Divisor	Baud rate	Divisor
50	2304	3600	32
75	1536	4800	24
150	768	7200	16
300	384	9600	12
600	192	19200	6
1200	96	38400	3
1800	64	57600	2
2400	48	115200	1

BASE+1	Interrupt Status Register (ISR) when DLAB=0	
Bit 0	Enable received-data-available interrupt	
bit 1	Enable transmitter-holding-register-empty interrupt	
bit 2	Enable receiver-line-status interrupt	
bit 3	Enable modem-status interrupt	

BASE+2	FIFO Control Register (FCR)		
bit 0	Enable transmit and receive FIFOs		
bit 1	Clear contents of receive FIFO		
bit 2	Clear contents of transmit FIFO		
bits 6-7	Set trigger level fro receiver FIFO interrupt.		
Bit 7	Bit 6	FIFO trigger level	
0	0	01	
0	1	04	
1	0	08	
1	1	14	

BASE+3	Line Control Register (LCR)		
bit 0	Word length select bit 0		
bit 1	Word length select bit 1		
Bit 1	Bit 0	Word length (bits)	
0	0	5	
0	1	6	
1	0	7	
1	1	8	
bit 2	Number of stop bits		
bit 3	Parity enable		
bit 4	even parity select		
bit 5	Stick parity		
bit 6	Set break		
bit 7	Divisor Latch Access Bit (DLAB)		

BASE+4	Modem Control Register (MCR)
	bit 0 DTR
	bit 1 RTS
BASE+5	Line Status Register (LSR)
	bit 0 Receiver data ready
	bit 1 Overrun error
	bit 2 Parity error
	bit 3 Framing error
	bit 4 Break interrupt
	bit 5 Transmitter holding register empty
	bit 6 Transmitter shift register empty
	bit 7 At least one parity error, framing error or break indication in the FIFO
BASE+6	Modem Status Register MSR)
	bit 0 Delta CTS
	bit 1 Delta DSR
	bit 2 Trailing edge ring indicator
	bit 3 Delta received line signal detect
	bit 4 CTS
	bit 5 DSR
	bit 6 RI
	bit 7 Received line signal detect
BASE+7	Temporary data register

Programming example

The following C example shows how to program the PCL-743/745 registers directly. It uses I/O ports hex 3F8 and 2F8 to test the PCL-743/745 send, receive and FIFO functions.

```
/
*****/
/* Program: DEM001.C
*/
/* Description:
*/
/* Sends a string from COM1 to COM2 then reads
*/
/* it back from COM2 and displays it on the
*/
/* screen. It uses direct register control.
*/
/
*****/

# include <dos.h>
# include <io.h>
# include <stdio.h>
# include <conic.h>

/* Timeout value in seconds */
#define TIME_OUT          10000
static int base0=0x3f8; /* Port 1 base address
(COM1)*/
static int base1=0x2f8; /*Port 2 base address
(COM2)*/

static char      rec[16]; /* Receive buffer */
static char      CMD[16]; /* Command buffer */

Void main()
{
    int i;
    timeout; /*      counter for timeout */
    char flag;
```

```

    /* Set up Port 1 (COM1) */
    outport((base0+2),0xc9);      /*      Enable FIFO
*/
    outp(base1+3,0x80);          /*      Set DLAB=1      */

    /*Set bps = 115200          */
    outp(base0 ,0x01); outp(base0+1,0x00);

    outp(base0+3,0x03); /* set data=8; stop=1; no
parity*/
    outp(base0+1,0x00); /* disable interrupt */

    /* (Set up) Port 2 (COM2)    */
    outport((base1+2),0xc9);      /*      Enable FIFO
*/
    outp(base1+3,0x80);          /*      Set DLAB=1      */

/* Set bps = 115,200 */
    outp(base1 ,0x01); outp(base1+1,0);

    outp(base1+3,0x03); /* Set data=8; stop=1; no pari-
ty*/
    outp(base1+1,0x00); /* Disable interrupt */

    printf(:"\nEnter string (max 15 char) or Q to
quit:");
    gets(cmd);

while (cmd{0} != 'q' && cmd[0] != 'Q')
{
    /* Send string on Port 1 (COM1)    */
    i=0;
    cmd[strlen(cmd)] = 0x0d;
    flag=1;
    while (flag)
    {
        outportb(base0,cmd[i]); /* send data */
        if (cmd[i] == 0x0d;

```

```

        */
        if (cmd[i] == 0x0d)
            flag=0
            i++;
    }

    /* Receive data on Port 2 (COM2) */
    i=0;
    flag=1;
    timeout=TIME_OUT;
    while (flag)
    {
        rec[i]=inportb(base1); /* Receive data*/
        if (rec[i] == 0x0d)
        {
            rec[i+1]='\0';
            flag=0;
            printf("\nReceived data: %s\n", rec);
        }
        i++;
    }
    else
    {
        /*I Check timeout */
        timeout--;
        if(timeout == 0)
        {
            flag=0;
            print("\nTimeout error\n");
        }
    }
} /*      End of receive data while()      */

    printf("\nEnter string (max 15 char) or Q to
quit:");
    gets(cmd);
} /*End of "Enter string"while() */

} /*End of main() */

```


APPENDIX **B**

**PC I/O address
reference**

The following table shows the I/O addresses commonly used by standard PC devices. Avoid these addresses when you select your port I/O base addresses.

I/O Address	Device
000 - 00F	DMA (8237A)
020 - 021	8259A IRQ Controller
040 - 043	8253/8254 Timer/Counter
060 - 063	PPI 8255A
070 - 071	Real-Time Clock
080 - 08F	DMA Page Register
0A0 - 0BF	8259A Interrupt Chip
0C0 - 0DF	Second DMA Controller 8237A
0F0 - 0FF	Math Coprocessor
1F0 - 1F8	AT Fixed Disk
200 - 20F	Game I/O
278 - 27F	Serial I/O Port #2
2F8 - 2FF	Serial Adaptor (COM 2)
320 - 32F	XT Fixed Disk
378 - 37F	Parallel Printer Adaptor
380 - 38F	SDLC Binary Communication Adaptor
3A0 - 3AF	Master Binary Communication Adaptor
3B0 - 3BF	Monochrome/Parallel Adaptor
3D0 - 3DF	Color Graphics Adaptor
3F0 - 3F7	Diskette Controller
3F8 - 3FF	Serial Adaptor (COM 1)