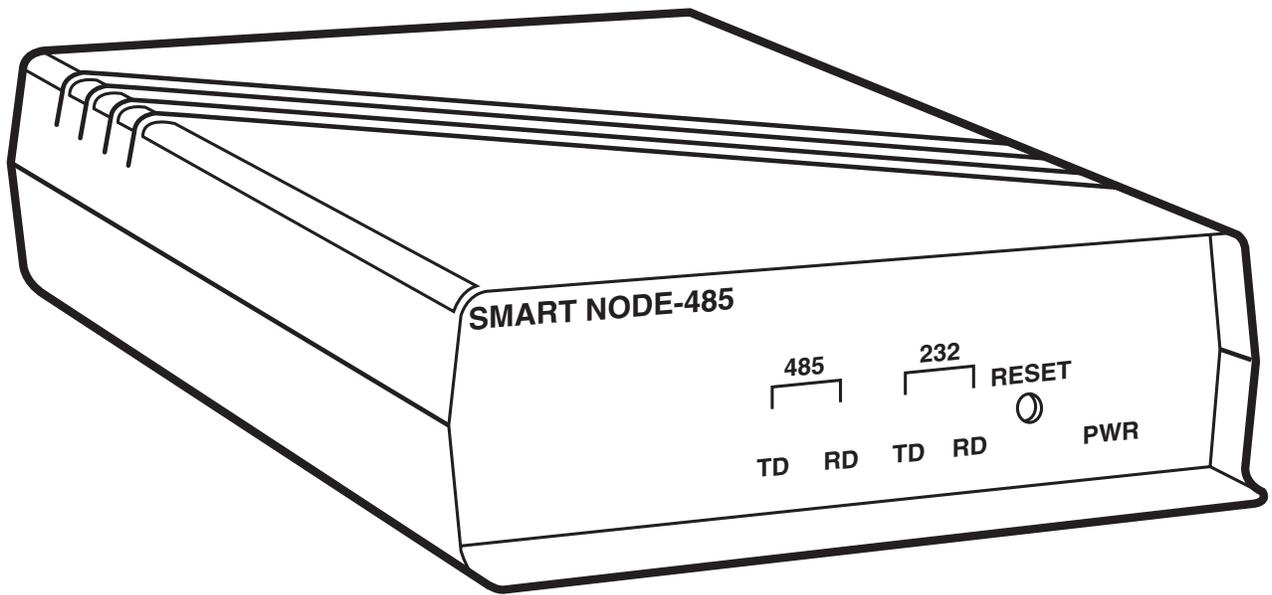




SmartNode



**CUSTOMER
SUPPORT
INFORMATION**

Order **toll-free** in the U.S. 24 hours, 7 A.M. Monday to midnight Friday: **877-877-BBOX**
FREE technical support, 24 hours a day, 7 days a week: Call **724-746-5500** or fax **724-746-0746**
Mail order: **Black Box Corporation**, 1000 Park Drive, Lawrence, PA 15055-1018
Web site: www.blackbox.com • E-mail: info@blackbox.com

**FEDERAL COMMUNICATIONS COMMISSION
RADIO FREQUENCY INTERFERENCE STATEMENT**

This equipment generates, uses and can radiate radio frequency energy and if not installed and used properly, that is in strict accordance with the manufacturer's instructions, may cause interference to radio communication. It has been tested and found to comply with the limits for a Class A computing device in accordance with the specifications in Subpart J of Part 15 of FCC rules, which are designed to provide reasonable protection against such interference when operated in a commercial environment. Operation of this equipment in a residential area is likely to cause interference, in which case the user at his own expense will be required to take whatever measures may be required to correct the interference.

Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

This digital apparatus does not exceed the Class A limits for Radio noise emission from digital apparatus set out in the Radio Interference Regulation of Industry Canada.

Le présent appareil numérique n'émet pas de bruits radioélectriques dépassant les limites applicables aux appareils numériques de la classe A prescrites dans le Règlement sur le brouillage radioélectrique édicté par Industrie Canada.

Contents

Section	Page
1. Specifications.....	3
2. Introduction	4
2.1 SmartNode Features	5
3. Operation	7
3.1 SmartNode Communication Record Format.....	7
3.2 Software Carrier-Detect Circuit.....	8
3.3 SmartNode Communications Protocols	8
3.4 Using the SmartNode as an Interface Converter in Front of a PC.....	10
4. Installing the SmartNode.....	11
4.1 DIP Switch Options.....	11
Appendix A: Sample Programs.....	22
Program 1: GWBASIC.....	22
Program 2: Turbo C	24
Appendix B: RS-232 and RS-485 Pinning	31
Appendix C: Testing the Units	32

Figures

2-1 Multidrop Network of RS-232 devices.....	4
2-2 Mixed Network of RS-485 and RS-232 devices.	5
3-1 SmartNode Record Format	7
4-1 SmartNode Component Diagram.....	11

Tables

4-1 Switch 1: Port A (RS-485 Options)	11
4-2 Switch 2: Port B (RS-232 Options)	12
4-3 Switch 3: Port A (RS-485 Options)	13
4-4 Switch 4: Port B(RS-232 Options)	14
4-5 Switch 5: Setting the SmartNode Address	15
4-6 Switch 7: Settings for RS-232 Port B.....	24
B-1 RS-232 and RS-485 Connector Pinning	31
B-2 RS-485 Terminal Block Layout	31

1. Specifications

Speed —	50 bps - 38.4 Kbps
Connectors —	(1) DB9 female DTE (1) 4-wire terminal block
Indicators —	Both ports: RD, TD
Leads supported —	RS-232: 1 through 9 (see chart for description)
Mean time between failure —	50,000 hours
Environmental considerations —	Storage temperature range: -4° - 158° F (-20° - 70° C) Operating temperature range: 32° - 122° F (0° - 50° C)
Maximum altitude —	10,000 feet
Power —	Primary: 115 VAC ±10% /60 Hz model or 230 VAC ±10%/50 Hz, 11 watts Secondary: 17 VAC center tap, 750 mA
	Connector (P1) pinning:
	1 frame ground
	2 secondary
	3 secondary
	4 signal ground
Size —	Standalone: 1.8"H x 5.5"W x 8.5"D (4.6 x 14 x 21.6 cm) Printed circuit board:4.5"H x 7.3"L (11.4 x 18.4 cm)
Weight —	2 lb. (0.9 Kg)

2. Introduction

SmartNode interfaces an asynchronous RS-232 device to an RS-485 multidrop ('daisychained') network. The SmartNode will add intelligence to a nonaddressable RS-232 device, so the nonaddressable device can communicate with the master device on the network without interfering with other slave and master communications.

The master device on the network controls when the nonaddressable RS-232 device will receive information and also when the slave can transmit information back to the master. The master port accomplishes this by following a unique protocol that the SmartNodes are programmed to understand.

The master device typically is either a programmable logic controller (PLC) with an RS-485 interface or a PC using an RS-232 \leftrightarrow RS-485 interface converter. Refer to figures 2-1 and 2-2 below for sample SmartNode applications.

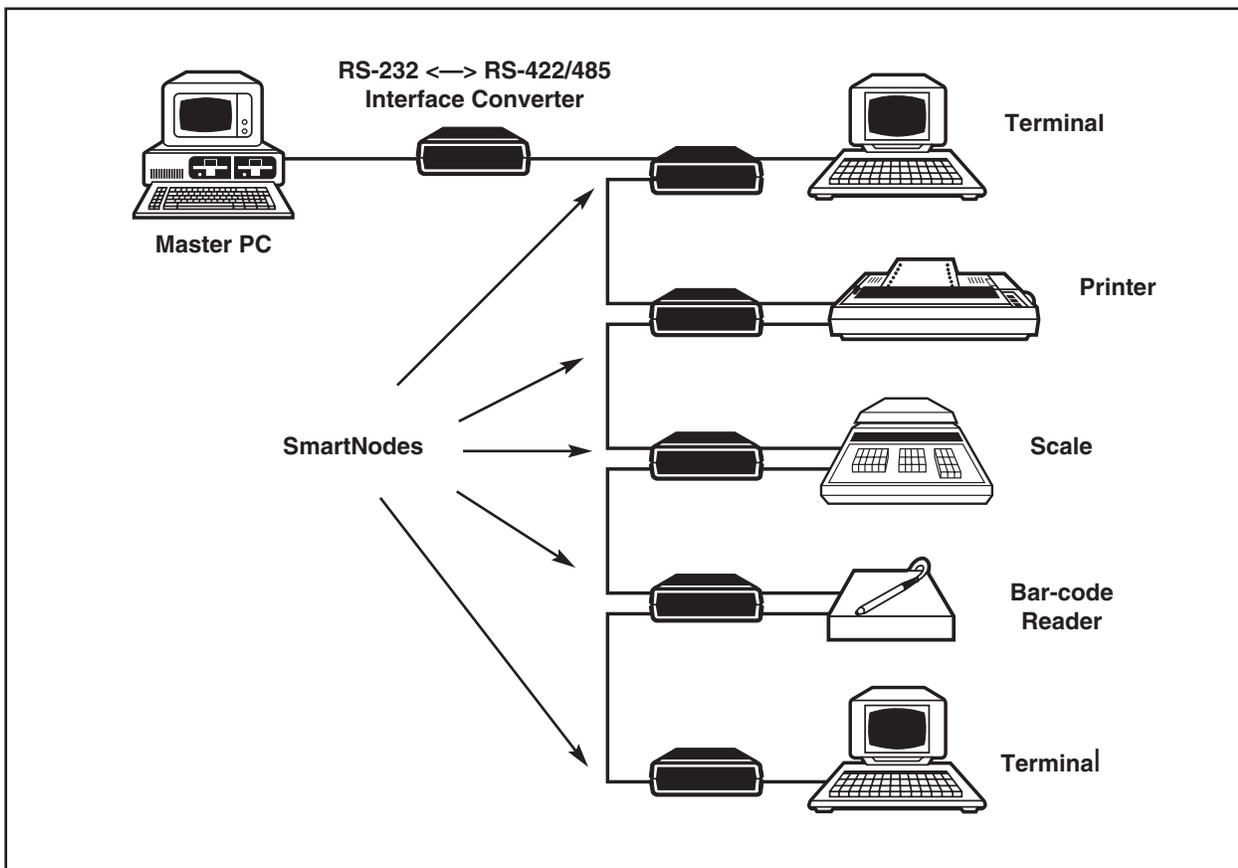


Figure 2-1. Multidrop Network of RS-232 Devices

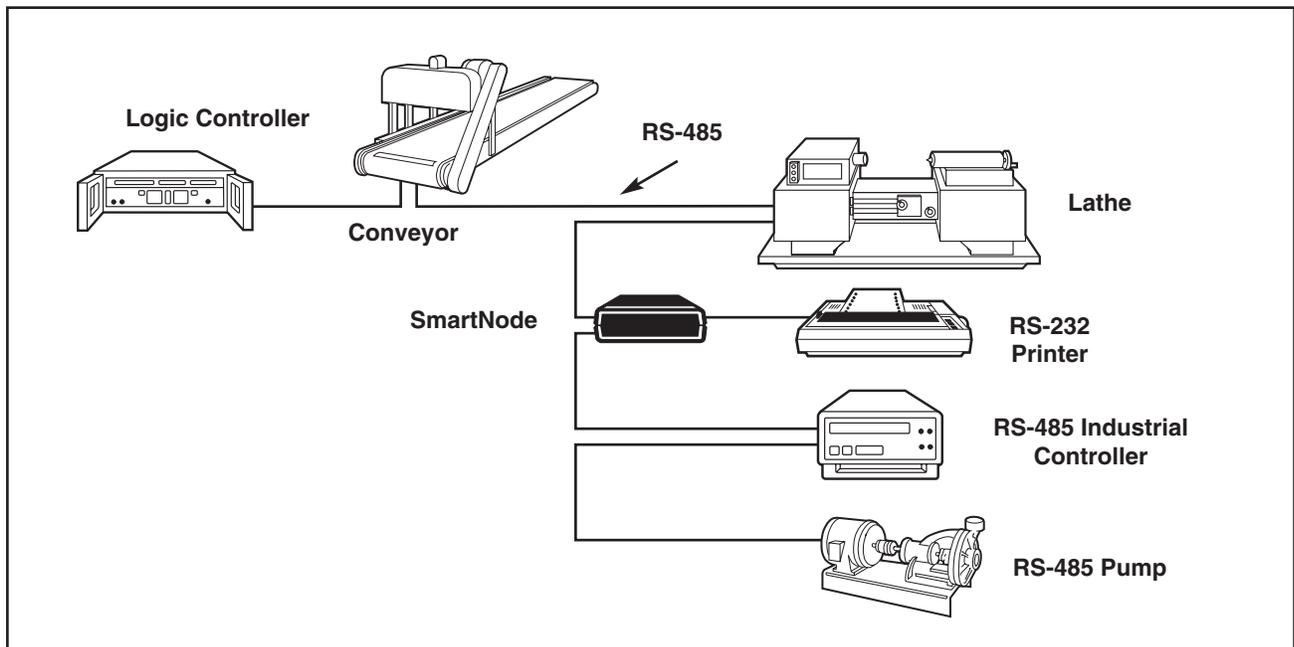


Figure 2-2 Mixed Network of RS-485 and RS-232 Devices

2.1 SmartNode Features

1. Because your RS-232 devices are connected to an RS-485 network through the SmartNodes you receive all the benefits of a multidrop line.
 - a). The RS-485 specification states that an RS-485 network can support up to 64 unit loads. The SmartNode allows a maximum of 256 addressable units on a network (three of those addresses are reserved: 0, 7E and 7F are reserved by the SmartNode protocol). The actual number of devices that can be attached to the network depends upon the total unit load of each device. In order to increase the number of units on a network beyond 32, repeaters will be necessary to maintain the communication signal strength.
 - b). RS-485 supports transmission distances up to 5,000 feet.
 - c). Using four-wire or two-wire channel reduces cable costs.
2. The SmartNode contains 8K of on board memory for buffering/spooling, .
3. The SmartNode gives you RS-232 flow control options for your nonaddressable devices.
- 4 Your nonaddressable RS-232 device can run a different word structure and baud rate than the RS-485 network.
- 5 `Built in' optional terminating resistors for the RS-485 receiver of any SmartNode that is placed on the end of the daisychain. For terminating the line, place switch S8 to the ON position.

Note: Only the last device (the one farthest from the master) should be terminated.

3. Operation

This section describes the rules for SmartNode communications.

3.1 SmartNode Communication Record Format

The following diagram is an example of how the master and slave nodes communicate on the network. Note that the TRAILER is optional; it can be replaced by any of the commands shown as options in Figure 3-1, and which are explained in Section 3.3 of this manual.

NOTE: The RS-485 driver must be enabled at least 50 milliseconds before HEADER string is sent by the host and must be disabled less than 50 milliseconds after sending the TRAILER string.

The length of time between HEADER and TRAILER is irrelevant; as long as the transmitting node maintains control of the line the listening node will not receive garbage data.

HEADER,	ADDRESS,	DATA (optional),	TRAILER (Option 1)
			->Trailer = the following four character
			ASCII string:]!\$@ (hex: 5D,21,24,40)
			Option 2: HEADER, ADDRESS (for different
			node)
			Option 3: HEADER, ADDRESS (7E hex for
			deselect)
		->	Asynchronous data for destination device (Note:
			SmartNodes pass ONLY this data section to the
			attached RS-232 device.)
	->		Address of destination node (0 thru 255 decimal or hexadecimal
			equivalent)
			SMARTNODE:
			address 00 hex - master address transmitted by all SmartNodes
			MASTER OPTIONS:
			address 7F hex - ALL SELECT - used by the master node to broadcast
			data to all SmartNodes
			address 7E hex - DESELECT - used by the master node to stop
			the previously addressed Smart Node(s) from transmitting over the
			multidrop network
			addresses 01-7D NODE SELECT - used by the master node
			and 80-FF to select a unique SmartNode
			->The following four character ASCII string]&#/ (hex: 7D,26,23,2F)

Figure 3-1. SmartNode Record Format

3.2 Software Carrier-Detect Circuit

The header and trailer strings used in the SmartNode communications record are necessary to complete a `SOFTWARE CARRIER DETECT` circuit. The header and trailer strings are the only way the SmartNode units can determine when the data received over the network is valid or invalid. Only data received between the header and trailer strings is considered valid. The ASCII characters chosen for these strings are unlikely to occur in other network traffic. Master node communication with non-SmartNode slaves can occur over the network as long as the communication does not include the unique header and trailer strings defined by the SmartNode protocol.

3.3 SmartNode Communication Protocol

This section describes the rules by which SmartNode communicates.

Initial 'Listen Only' State

After a power up or manual reset only the host device is allowed to transmit data over the multidrop network. SmartNodes will listen for a valid header string on the RS-485 network; they may also buffer and hold any data received from the attached RS-232 device unless configured to only buffer this data when the SmartNode is selected to transmit by the master.

Transmitting Data to SmartNodes

When a SmartNode receives a valid header string followed by its own unique address or the the `All Select` address (7F hex) it will begin to buffer any RS-485 data that follows. This information will be retransmitted to the attached RS-232 device as it is received using the baud rate, word structure, and flow control optioned for that device. The SmartNode will stop buffering RS-485 data when one of the following events happen:

1. The master device transmits a predefined trailer string. Note that the trailer string is not buffered. (This is Option 1 from Figure 3-1: HEADER, ADDRESS, DATA, TRAILER.)

NOTE: Only a record following the format of Option 1 gives the SmartNode permission to transmit RS-232 data to the master device.

2. The master device transmits a valid header string followed by a different slave's address. Note that the header string is not buffered. (Option 2: HEADER, ADDRESS, DATA, HEADER, ADDRESS (address of different node)...))
3. The master device transmits a valid header string followed by the `Deselect` address, 7E hex. (Option 3: HEADER, ADDRESS, DATA, HEADER, ADDRESS (7E))
4. A reset of the SmartNode (power-up or manual reset).

Receiving Data From SmartNodes

The master device gives a slave SmartNode permission to transmit by sending it a complete SmartNode communication record containing that node's assigned address. A record is complete when it ends with a valid TRAILER string. The data section in this host message is optional, but if it is present it will be passed onto the RS-232 device as described above. The SmartNode will not activate the RS-485 driver until it has something to transmit over the RS-485 network. The SmartNode will transmit a valid header string followed by the destination address of 00 hex. Once a SmartNode has been given permission to transmit to the master it can continue to transmit until one of the following events happen:

1. The master device transmits the 'Deselect' address following a valid header string (4-wire applications only).
2. The SmartNode's 'NO TRANSMIT' timeout occurs. This timeout is enabled via DIP switch selection. The timeout period ranges from 1 second to 4 minutes. If the SmartNode does have any slave data to retransmit over the network for the selected timeout period the SmartNode will terminate its transmission.
3. The SmartNode's 'SESSION' timeout occurs. This timeout is enabled via DIP switch selection. The timeout period ranges from 5 seconds to 5 minutes. If this timeout is enabled, the SmartNode's transmission over the multidrop network will be terminated after the selected timeout period expires from when the unit is given permission to transmit (even if there is additional buffered data to transmit)
4. A reset of the SmartNode (power up or manual reset).

The SmartNode will terminate its transmission by sending the valid trailer string and then disabling its RS-485 driver in all the above cases except when the unit is reset.

Special Requirements

To avoid collisions on a 2-wire network, the SmartNodes will not begin transmitting over the multidrop network until they have received the valid four-character trailer string that terminates a communication record. Master nodes should also be programmed with this protocol if using a 2-wire network.

The master node should enable the transmitter on its RS-485 line driver for a minimum of 50 milliseconds before transmitting the first byte of the header string. The SmartNode will follow this procedure when it is selected to transmit. This delay reduces the likelihood that the header string is garbled due to random noise on the RS-485 network.

The RS-485 transmitter of the line driver should be driven by the master node during the entire SmartNode communication record (even during intercharacter gaps of the data) to assure that noise is not received by the SmartNodes.

The master node's RS-485 line driver's transmitter should also be released within 50 milliseconds after completely transmitting the last byte of the trailer string. This will prevent two nodes from transmitting data on the multidrop line at the same time in 2-wire networks.

3.4 Using the SmartNode as an Interface Converter in Front of a PC.

NOTE: A line driver is best suited to interface a PC to a SmartNode network. However, the following information has been listed to offer you another option.

The SmartNode's address can be set to the reserved host address (zero). This allows the unit to be an interface converter in front of your RS-232 host. The unit is not a transparent interface converter because the SmartNode still performs the following functions:

- header insertion - before every message transmitted over the network
- trailer insertion - when the no-transmit timeout occurs.
- header/trailer deletion - any messages containing the host address zero will be passed through the unit with the header, address, and trailer removed from the data stream

The SmartNode will do the following additional functions when it is configured for host address zero:

- the unit is always allowed to transmit over the network (the protocol to transmit only after receiving the address and complete trailer string no longer applies)
- the unit will pass the first byte of data received from the RS-232 device as the address behind the header (it will not insert address zero).
- the unit will always buffer RS-232 data (the option to buffer data only when addressed in not applicable)

4. Installing the SmartNode

This section will help you to set up the SmartNode for your application. Refer to Figure 4-1 for the location of components referenced throughout this manual.

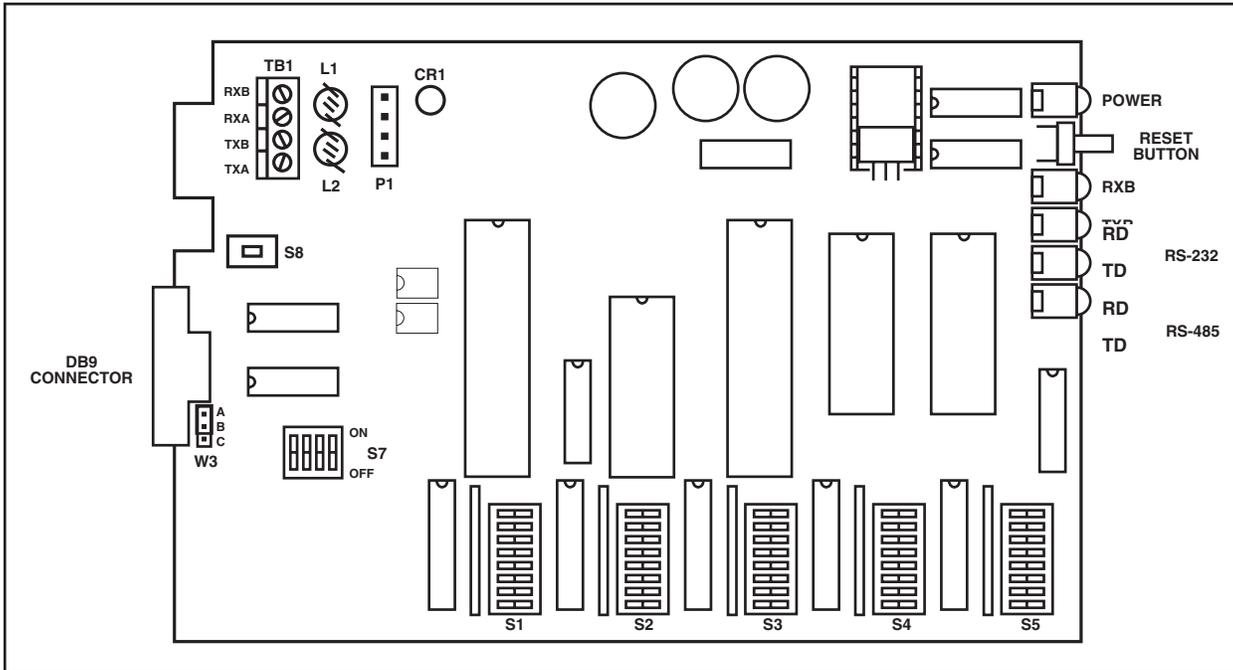


Figure 4-1. SmartNode Component Diagram

4.1 Dipswitch Options

Table 4-1. Switch 1: Port A (RS-485 Options)

Word Structure

	1	2
1 stop bit	off	off
1.5 stop bits	on	off
1 stop bit	off	on
2 stop bits	on	on

Parity

	3
odd parity	off
even parity	on

Table 4-1. Switch 1: Port A (RS-485 Options) (continued)

	4		
parity disabled	off		
parity enabled	on		
Data Bits			
		5	6
8 data bits		off	off
7 data bits		off	on
reserved for future use		on	off
reserved for future use		on	on
		7	8
Reserved for Future Use			
		*	*

Table 4-2. Switch 2: Port B (RS-232 Options)

Word Structure			
		1	2
1	stop bit	off	off
1.5	stop bit	on	off
1	stop bit	off	on
2	stop bits	on	on
Parity			
			3
odd parity			off
even parity			on
			4
parity disabled			off
parity enabled			on
Data Bits			
		5	6
8 data bits		off	off
7 data bits		off	on
reserved for future use		on	off
reserved for future use		on	on

Table 4-2. Switch 2: Port B (RS-232 Options) (continued)

Flow Control

	7	8
Hardware (CTS/DTR)	off	off
ENQ/ACK	on	off
XON/XOFF	off	on
XON/XOFF (Transmit XON on reset)	on	on

Table 4-3. Switch 3: Port A (RS-485 Options)

Baud Rate	Position			
	1	2	3	4
38,400	off	off	off	off
19,200	on	off	off	off
9,600	off	on	off	off
4,800	on	on	off	off
2,400	off	off	on	off
1,200	on	off	on	off
600	off	on	on	off
300	on	on	on	off
200	off	off	off	on
150	on	off	off	on
110	off	on	off	on
100	on	on	off	on
74.42	off	off	on	on
55.58	on	off	on	on
50	off	on	on	on
45.5	on	on	on	on

Reserved for Future Use

5
*

‘Session’ Timeout on RS-485 Network to Terminate Transmission

	6	7	8
disabled	off	off	off
5 seconds	on	off	off
10 seconds	off	on	off
30 seconds	on	on	off
1 minute	off	off	on
3 minutes	on	off	on
4 minutes	off	on	on
5 minute	on	on	on

Table 4-4. Switch 4: Port B (RS-232 Options)

Baud Rate

	1	2	3	4
38,400	off	off	off	off
19,200	on	off	off	off
9,600	off	on	off	off
4,800	on	on	off	off
2,400	off	off	on	off
1,200	on	off	on	off
600	off	on	on	off
300	on	on	on	off
200	off	off	off	on
150	on	off	off	on
110	off	on	off	on
100	on	on	off	on
74.42	off	off	on	on
55.58	on	off	on	on
50	off	on	on	on
45.5	on	on	on	on

Buffer Option

	5
always buffer RS-232 receive data	off
only buffer when selected to transmit by the master node	on

'NO TRANSMIT' Time out on RS-485 Network to Terminate Transmission

	6	7	8
disabled	off	off	off
1 second	on	off	off
5 seconds	off	on	off
10 seconds	on	on	off
30 seconds	off	off	on
1 minute	on	off	on
3 minutes	off	on	on
4 minutes	on	on	on

Table 4-5. Switch 5: Setting the SmartNode Address

ADDRESS		1	2	3	4	5	6	7	8
255 DECIMAL	FF HEX	ON	ON	ON	ON	ON	ON	ON	ON
254 DECIMAL	FE HEX	OFF	ON	ON	ON	ON	ON	ON	ON
253 DECIMAL	FD HEX	ON	OFF	ON	ON	ON	ON	ON	ON
252 DECIMAL	FC HEX	OFF	OFF	ON	ON	ON	ON	ON	ON
251 DECIMAL	FB HEX	ON	ON	OFF	ON	ON	ON	ON	ON
250 DECIMAL	FA HEX	OFF	ON	OFF	ON	ON	ON	ON	ON
249 DECIMAL	F9 HEX	ON	OFF	OFF	ON	ON	ON	ON	ON
248 DECIMAL	F8 HEX	OFF	OFF	OFF	ON	ON	ON	ON	ON
247 DECIMAL	F7 HEX	ON	ON	ON	OFF	ON	ON	ON	ON
246 DECIMAL	F6 HEX	OFF	ON	ON	OFF	ON	ON	ON	ON
245 DECIMAL	F5 HEX	ON	OFF	ON	OFF	ON	ON	ON	ON
244 DECIMAL	F4 HEX	OFF	OFF	ON	OFF	ON	ON	ON	ON
243 DECIMAL	F3 HEX	ON	ON	OFF	OFF	ON	ON	ON	ON
242 DECIMAL	F2 HEX	OFF	ON	OFF	OFF	ON	ON	ON	ON
241 DECIMAL	F1 HEX	ON	OFF	OFF	OFF	ON	ON	ON	ON
240 DECIMAL	F0 HEX	OFF	OFF	OFF	OFF	ON	ON	ON	ON
239 DECIMAL	EF HEX	ON	ON	ON	ON	OFF	ON	ON	ON
238 DECIMAL	EE HEX	OFF	ON	ON	ON	OFF	ON	ON	ON
237 DECIMAL	ED HEX	ON	OFF	ON	ON	OFF	ON	ON	ON
236 DECIMAL	EC HEX	OFF	OFF	ON	ON	OFF	ON	ON	ON
235 DECIMAL	EB HEX	ON	ON	OFF	ON	OFF	ON	ON	ON
234 DECIMAL	EA HEX	OFF	ON	OFF	ON	OFF	ON	ON	ON
233 DECIMAL	E9 HEX	ON	OFF	OFF	ON	OFF	ON	ON	ON
232 DECIMAL	E8 HEX	OFF	OFF	OFF	ON	OFF	ON	ON	ON
231 DECIMAL	E7 HEX	ON	ON	ON	OFF	OFF	ON	ON	ON
230 DECIMAL	E6 HEX	OFF	ON	ON	OFF	OFF	ON	ON	ON
229 DECIMAL	E5 HEX	ON	OFF	ON	OFF	OFF	ON	ON	ON
228 DECIMAL	E4 HEX	OFF	OFF	ON	OFF	OFF	ON	ON	ON
227 DECIMAL	E3 HEX	ON	ON	OFF	OFF	OFF	ON	ON	ON
226 DECIMAL	E2 HEX	OFF	ON	OFF	OFF	OFF	ON	ON	ON
225 DECIMAL	E1 HEX	ON	OFF	OFF	OFF	OFF	ON	ON	ON
224 DECIMAL	E0 HEX	OFF	OFF	OFF	OFF	OFF	ON	ON	ON
223 DECIMAL	DF HEX	ON	ON	ON	ON	ON	OFF	ON	ON
222 DECIMAL	DE HEX	OFF	ON	ON	ON	ON	OFF	ON	ON
221 DECIMAL	DD HEX	ON	OFF	ON	ON	ON	OFF	ON	ON
220 DECIMAL	DC HEX	OFF	OFF	ON	ON	ON	OFF	ON	ON
219 DECIMAL	DB HEX	ON	ON	OFF	ON	ON	OFF	ON	ON
218 DECIMAL	DA HEX	OFF	ON	OFF	ON	ON	OFF	ON	ON
217 DECIMAL	D9 HEX	ON	OFF	OFF	ON	ON	OFF	ON	ON

Table 4-5. Switch 5: Setting the SmartNode Address (continued)

ADDRESS		1	2	3	4	5	6	7	8
216 DECIMAL	D8 HEX	OFF	OFF	OFF	ON	ON	OFF	ON	ON
215 DECIMAL	D7 HEX	ON	ON	ON	OFF	ON	OFF	ON	ON
214 DECIMAL	D6 HEX	OFF	ON	ON	OFF	ON	OFF	ON	ON
213 DECIMAL	D5 HEX	ON	OFF	ON	OFF	ON	OFF	ON	ON
212 DECIMAL	D4 HEX	OFF	OFF	ON	OFF	ON	OFF	ON	ON
211 DECIMAL	D3 HEX	ON	ON	OFF	OFF	ON	OFF	ON	ON
210 DECIMAL	D2 HEX	OFF	ON	OFF	OFF	ON	OFF	ON	ON
209 DECIMAL	D1 HEX	ON	OFF	OFF	OFF	ON	OFF	ON	ON
208 DECIMAL	D0 HEX	OFF	OFF	OFF	OFF	ON	OFF	ON	ON
207 DECIMAL	CF HEX	ON	ON	ON	ON	OFF	OFF	ON	ON
206 DECIMAL	CE HEX	OFF	ON	ON	ON	OFF	OFF	ON	ON
205 DECIMAL	CD HEX	ON	OFF	ON	ON	OFF	OFF	ON	ON
204 DECIMAL	CC HEX	OFF	OFF	ON	ON	OFF	OFF	ON	ON
203 DECIMAL	CB HEX	ON	ON	OFF	ON	OFF	OFF	ON	ON
202 DECIMAL	CA HEX	OFF	ON	OFF	ON	OFF	OFF	ON	ON
201 DECIMAL	C9 HEX	ON	OFF	OFF	ON	OFF	OFF	ON	ON
200 DECIMAL	C8 HEX	OFF	OFF	OFF	ON	OFF	OFF	ON	ON
199 DECIMAL	C7 HEX	ON	ON	ON	OFF	OFF	OFF	ON	ON
198 DECIMAL	C6 HEX	OFF	ON	ON	OFF	OFF	OFF	ON	ON
197 DECIMAL	C5 HEX	ON	OFF	ON	OFF	OFF	OFF	ON	ON
196 DECIMAL	C4 HEX	OFF	OFF	ON	OFF	OFF	OFF	ON	ON
195 DECIMAL	C3 HEX	ON	ON	OFF	OFF	OFF	OFF	ON	ON
194 DECIMAL	C2 HEX	OFF	ON	OFF	OFF	OFF	OFF	ON	ON
193 DECIMAL	C1 HEX	ON	OFF	OFF	OFF	OFF	OFF	ON	ON
192 DECIMAL	C0 HEX	OFF	OFF	OFF	OFF	OFF	OFF	ON	ON
191 DECIMAL	BF HEX	ON	ON	ON	ON	ON	ON	OFF	ON
190 DECIMAL	BE HEX	OFF	ON	ON	ON	ON	ON	OFF	ON
189 DECIMAL	BD HEX	ON	OFF	ON	ON	ON	ON	OFF	ON
188 DECIMAL	BC HEX	OFF	OFF	ON	ON	ON	ON	OFF	ON
187 DECIMAL	BB HEX	ON	ON	OFF	ON	ON	ON	OFF	ON
186 DECIMAL	BA HEX	OFF	ON	OFF	ON	ON	ON	OFF	ON
185 DECIMAL	B9 HEX	ON	OFF	OFF	ON	ON	ON	OFF	ON
184 DECIMAL	B8 HEX	OFF	OFF	OFF	ON	ON	ON	OFF	ON
183 DECIMAL	B7 HEX	ON	ON	ON	OFF	ON	ON	OFF	ON
182 DECIMAL	B6 HEX	OFF	ON	ON	OFF	ON	ON	OFF	ON
181 DECIMAL	B5 HEX	ON	OFF	ON	OFF	ON	ON	OFF	ON
180 DECIMAL	B4 HEX	OFF	OFF	ON	OFF	ON	ON	OFF	ON
179 DECIMAL	B3 HEX	ON	ON	OFF	OFF	ON	ON	OFF	ON
178 DECIMAL	B2 HEX	OFF	ON	OFF	OFF	ON	ON	OFF	ON
177 DECIMAL	B1 HEX	ON	OFF	OFF	OFF	ON	ON	OFF	ON
176 DECIMAL	B0 HEX	OFF	OFF	OFF	OFF	ON	ON	OFF	ON

Table 4-5. Switch 5: Setting the SmartNode Address (continued)

ADDRESS		1	2	3	4	5	6	7	8
175 DECIMAL	AF HEX	ON	ON	ON	ON	OFF	ON	OFF	ON
174 DECIMAL	AE HEX	OFF	ON	ON	ON	OFF	ON	OFF	ON
173 DECIMAL	AD HEX	ON	OFF	ON	ON	OFF	ON	OFF	ON
172 DECIMAL	AC HEX	OFF	OFF	ON	ON	OFF	ON	OFF	ON
171 DECIMAL	AB HEX	ON	ON	OFF	ON	OFF	ON	OFF	ON
170 DECIMAL	AA HEX	OFF	ON	OFF	ON	OFF	ON	OFF	ON
169 DECIMAL	A9 HEX	ON	OFF	OFF	ON	OF	ON	OFF	ON
168 DECIMAL	A8 HEX	OFF	OFF	OFF	ON	OFF	ON	OFF	ON
167 DECIMAL	A7 HEX	ON	ON	ON	OFF	OFF	ON	OFF	ON
166 DECIMAL	A6 HEX	OFF	ON	ON	OFF	OFF	ON	OFF	ON
165 DECIMAL	A5 HEX	ON	OFF	ON	OFF	OFF	ON	OFF	ON
164 DECIMAL	A4 HEX	OFF	OFF	ON	OFF	OFF	ON	OFF	ON
163 DECIMAL	A3 HEX	ON	ON	OFF	OFF	OFF	ON	OFF	ON
162 DECIMAL	A2 HEX	OFF	ON	OFF	OFF	OFF	ON	OFF	ON
161 DECIMAL	A1 HEX	ON	OFF	OFF	OFF	OFF	ON	OFF	ON
160 DECIMAL	A0 HEX	OFF	OFF	OFF	OFF	OFF	ON	OFF	ON
159 DECIMAL	9F HEX	ON	ON	ON	ON	ON	OFF	OFF	ON
158 DECIMAL	9E HEX	OFF	ON	ON	ON	ON	OFF	OFF	ON
157 DECIMAL	9D HEX	ON	OFF	ON	ON	ON	OFF	OFF	ON
156 DECIMAL	9C HEX	OFF	OFF	ON	ON	ON	OFF	OFF	ON
155 DECIMAL	9B HEX	ON	ON	OFF	ON	ON	OFF	OFF	ON
154 DECIMAL	9A HEX	OFF	ON	OFF	ON	ON	OFF	OFF	ON
153 DECIMAL	99 HEX	ON	OFF	OFF	ON	ON	OFF	OFF	ON
152 DECIMAL	98 HEX	OFF	OFF	OFF	ON	ON	OFF	OFF	ON
151 DECIMAL	97 HEX	ON	ON	ON	OFF	ON	OFF	OFF	ON
150 DECIMAL	96 HEX	OFF	ON	ON	OFF	ON	OFF	OFF	ON
149 DECIMAL	95 HEX	ON	OFF	ON	OFF	ON	OFF	OFF	ON
148 DECIMAL	94 HEX	OFF	OFF	ON	OFF	ON	OFF	OFF	ON
147 DECIMAL	93 HEX	ON	ON	OFF	OFF	ON	OFF	OFF	ON
146 DECIMAL	92 HEX	OFF	ON	OFF	OFF	ON	OFF	OFF	ON
145 DECIMAL	91 HEX	ON	OFF	OFF	OFF	ON	OFF	OFF	ON
144 DECIMAL	90 HEX	OFF	OFF	OFF	OFF	ON	OFF	OFF	ON
143 DECIMAL	8F HEX	ON	ON	ON	ON	OFF	OFF	OFF	ON
142 DECIMAL	8E HEX	OFF	ON	ON	ON	OFF	OFF	OFF	ON
141 DECIMAL	8D HEX	ON	OFF	ON	ON	OFF	OFF	OFF	ON
140 DECIMAL	8C HEX	OFF	OFF	ON	ON	OFF	OFF	OFF	ON
139 DECIMAL	8B HEX	ON	ON	OFF	ON	OFF	OFF	OFF	ON
138 DECIMAL	8A HEX	OFF	ON	OFF	ON	OFF	OFF	OFF	ON
137 DECIMAL	89 HEX	ON	OFF	OFF	ON	OFF	OFF	OFF	ON
136 DECIMAL	88 HEX	OFF	OFF	OFF	ON	OFF	OFF	OFF	ON
135 DECIMAL	87 HEX	ON	ON	ON	OFF	OFF	OFF	OFF	ON

Table 4-5. Switch 5: Setting the SmartNode Address (continued)

ADDRESS		1	2	3	4	5	6	7	8
134 DECIMAL	86 HEX	OFF	ON	ON	OFF	OFF	OFF	OFF	ON
133 DECIMAL	85 HEX	ON	OFF	ON	OFF	OFF	OFF	OFF	ON
132 DECIMAL	84 HEX	OFF	OFF	ON	OFF	OFF	OFF	OFF	ON
131 DECIMAL	83 HEX	ON	ON	OFF	OFF	OFF	OFF	OFF	ON
130 DECIMAL	82 HEX	OFF	ON	OFF	OFF	OFF	OFF	OFF	ON
129 DECIMAL	81 HEX	ON	OFF	OFF	OFF	OFF	OFF	OFF	ON
128 DECIMAL	80 HEX	OFF	ON						
*127 DECIMAL	7F HEX	ON	OFF						
*126 DECIMAL	7E HEX	OFF	ON	ON	ON	ON	ON	ON	OFF
125 DECIMAL	7D HEX	ON	OFF	ON	ON	ON	ON	ON	OFF
124 DECIMAL	7C HEX	OFF	OFF	ON	ON	ON	ON	ON	OFF
123 DECIMAL	7B HEX	ON	ON	OFF	ON	ON	ON	ON	OFF
122 DECIMAL	7A HEX	OFF	ON	OFF	ON	ON	ON	ON	OFF
121 DECIMAL	79 HEX	ON	OFF	OFF	ON	ON	ON	ON	OFF
120 DECIMAL	78 HEX	OFF	OFF	OFF	ON	ON	ON	ON	OFF
119 DECIMAL	77 HEX	ON	ON	ON	OFF	ON	ON	ON	OFF
118 DECIMAL	76 HEX	OFF	ON	ON	OFF	ON	ON	ON	OFF
117 DECIMAL	75 HEX	ON	OFF	ON	OFF	ON	ON	ON	OFF
116 DECIMAL	74 HEX	OFF	OFF	ON	OFF	ON	ON	ON	OFF
115 DECIMAL	73 HEX	ON	ON	OFF	OFF	ON	ON	ON	OFF
114 DECIMAL	72 HEX	OFF	ON	OFF	OFF	ON	ON	ON	OFF
113 DECIMAL	71 HEX	ON	OFF	OFF	OFF	ON	ON	ON	OFF
112 DECIMAL	70 HEX	OFF	OFF	OFF	OFF	ON	ON	ON	OFF
111 DECIMAL	6F HEX	ON	ON	ON	ON	OFF	ON	ON	OFF
110 DECIMAL	6E HEX	OFF	ON	ON	ON	OFF	ON	ON	OFF
109 DECIMAL	6D HEX	ON	OFF	ON	ON	OFF	ON	ON	OFF
108 DECIMAL	6C HEX	OFF	OFF	ON	ON	OFF	ON	ON	OFF
107 DECIMAL	6B HEX	ON	ON	OFF	ON	OFF	ON	ON	OFF
106 DECIMAL	6A HEX	OFF	ON	OFF	ON	OFF	ON	ON	OFF
105 DECIMAL	69 HEX	ON	OFF	OFF	ON	OFF	ON	ON	OFF
104 DECIMAL	68 HEX	OFF	OFF	OFF	ON	OFF	ON	ON	OFF
103 DECIMAL	67 HEX	ON	ON	ON	OFF	OFF	ON	ON	OFF
102 DECIMAL	66 HEX	OFF	ON	ON	OFF	OFF	ON	ON	OFF
101 DECIMAL	65 HEX	ON	OFF	ON	OFF	OFF	ON	ON	OFF
100 DECIMAL	64 HEX	OFF	OFF	ON	OFF	OFF	ON	ON	OFF
99 DECIMAL	63 HEX	ON	ON	OFF	OFF	OFF	ON	ON	OFF
98 DECIMAL	62 HEX	OFF	ON	OFF	OFF	OFF	ON	ON	OFF
97 DECIMAL	61 HEX	ON	OFF	OFF	OFF	OFF	ON	ON	OFF
96 DECIMAL	60 HEX	OFF	OFF	OFF	OFF	OFF	ON	ON	OFF
95 DECIMAL	5F HEX	ON	ON	ON	ON	ON	OFF	ON	OFF
94 DECIMAL	5E HEX	OFF	ON	ON	ON	ON	OFF	ON	OFF

Table 4-5. Switch 5: Setting the SmartNode Address (continued)

ADDRESS		1	2	3	4	5	6	7	8
93 DECIMAL	5D HEX	ON	OFF	ON	ON	ON	OFF	ON	OFF
92 DECIMAL	5C HEX	OFF	OFF	ON	ON	ON	OFF	ON	OFF
91 DECIMAL	5B HEX	ON	ON	OFF	ON	ON	OFF	ON	OFF
90 DECIMAL	5A HEX	OFF	ON	OFF	ON	ON	OFF	ON	OFF
89 DECIMAL	59 HEX	ON	OFF	OFF	ON	ON	OFF	ON	OFF
88 DECIMAL	58 HEX	OFF	OFF	OFF	ON	ON	OFF	ON	OFF
87 DECIMAL	57 HEX	ON	ON	ON	OFF	ON	OFF	ON	OFF
86 DECIMAL	56 HEX	OFF	ON	ON	OFF	ON	OFF	ON	OFF
85 DECIMAL	55 HEX	ON	OFF	ON	OFF	ON	OFF	ON	OFF
84 DECIMAL	54 HEX	OFF	OFF	ON	OFF	ON	OFF	ON	OFF
83 DECIMAL	53 HEX	ON	ON	OFF	OFF	ON	OFF	ON	OFF
82 DECIMAL	52 HEX	OFF	ON	OFF	OFF	ON	OFF	ON	OFF
81 DECIMAL	51 HEX	ON	OFF	OFF	OFF	ON	OFF	ON	OFF
80 DECIMAL	50 HEX	OFF	OFF	OFF	OFF	ON	OFF	ON	OFF
79 DECIMAL	4F HEX	ON	ON	ON	ON	OFF	OFF	ON	OFF
78 DECIMAL	4E HEX	OFF	ON	ON	ON	OFF	OFF	ON	OFF
77 DECIMAL	4D HEX	ON	OFF	ON	ON	OFF	OFF	ON	OFF
76 DECIMAL	4C HEX	OFF	OFF	ON	ON	OFF	OFF	ON	OFF
75 DECIMAL	4B HEX	ON	ON	OFF	ON	OFF	OFF	ON	OFF
74 DECIMAL	4A HEX	OFF	ON	OFF	ON	OFF	OFF	ON	OFF
73 DECIMAL	49 HEX	ON	OFF	OFF	ON	OFF	OFF	ON	OFF
72 DECIMAL	48 HEX	OFF	OFF	OFF	ON	OFF	OFF	ON	OFF
71 DECIMAL	47 HEX	ON	ON	ON	OFF	OFF	OFF	ON	OFF
70 DECIMAL	46 HEX	OFF	ON	ON	OFF	OFF	OFF	ON	OFF
69 DECIMAL	45 HEX	ON	OFF	ON	OFF	OFF	OFF	ON	OFF
68 DECIMAL	44 HEX	OFF	OFF	ON	OFF	OFF	OFF	ON	OFF
67 DECIMAL	43 HEX	ON	ON	OFF	OFF	OFF	OFF	ON	OFF
66 DECIMAL	42 HEX	OFF	ON	OFF	OFF	OFF	OFF	ON	OFF
65 DECIMAL	41 HEX	ON	OFF	OFF	OFF	OFF	OFF	ON	OFF
64 DECIMAL	40 HEX	OFF	OFF	OFF	OFF	OFF	OFF	ON	OFF
63 DECIMAL	3F HEX	ON	ON	ON	ON	ON	ON	OFF	OFF
62 DECIMAL	3E HEX	OFF	ON	ON	ON	ON	ON	OFF	OFF
61 DECIMAL	3D HEX	ON	OFF	ON	ON	ON	ON	OFF	OFF
60 DECIMAL	3C HEX	OFF	OFF	ON	ON	ON	ON	OFF	OFF
59 DECIMAL	3B HEX	ON	ON	OFF	ON	ON	ON	OFF	OFF
58 DECIMAL	3A HEX	OFF	ON	OFF	ON	ON	ON	OFF	OFF
57 DECIMAL	39 HEX	ON	OFF	OFF	ON	ON	ON	OFF	OFF
56 DECIMAL	38 HEX	OFF	OFF	OFF	ON	ON	ON	OFF	OFF
55 DECIMAL	37 HEX	ON	ON	ON	OFF	ON	ON	OFF	OFF
54 DECIMAL	36 HEX	OFF	ON	ON	OFF	ON	ON	OFF	OFF
53 DECIMAL	35 HEX	ON	OFF	ON	OFF	ON	ON	OFF	OFF

Table 4-5. Switch 5: Setting the SmartNode Address (continued)

ADDRESS		1	2	3	4	5	6	7	8
52 DECIMAL	34 HEX	OFF	OFF	ON	OFF	ON	ON	OFF	OFF
51 DECIMAL	33 HEX	ON	ON	OFF	OFF	ON	ON	OFF	OFF
50 DECIMAL	32 HEX	OFF	ON	OFF	OFF	ON	ON	OFF	OFF
49 DECIMAL	31 HEX	ON	OFF	OFF	OFF	ON	ON	OFF	OFF
48 DECIMAL	30 HEX	OFF	OFF	OFF	OFF	ON	ON	OFF	OFF
47 DECIMAL	2F HEX	ON	ON	ON	ON	OFF	ON	OFF	OFF
46 DECIMAL	2E HEX	OFF	ON	ON	ON	OFF	ON	OFF	OFF
45 DECIMAL	2D HEX	ON	OFF	ON	ON	OFF	ON	OFF	OFF
44 DECIMAL	2C HEX	OFF	OFF	ON	ON	OFF	ON	OFF	OFF
43 DECIMAL	2B HEX	ON	ON	OFF	ON	OFF	ON	OFF	OFF
42 DECIMAL	2A HEX	OFF	ON	OFF	ON	OFF	ON	OFF	OFF
41 DECIMAL	29 HEX	ON	OFF	OFF	ON	OFF	ON	OFF	OFF
40 DECIMAL	28 HEX	OFF	OFF	OFF	ON	OFF	ON	OFF	OFF
39 DECIMAL	27 HEX	ON	ON	ON	OFF	OFF	ON	OFF	OFF
38 DECIMAL	26 HEX	OFF	ON	ON	OFF	OFF	ON	OFF	OFF
37 DECIMAL	25 HEX	ON	OFF	ON	OFF	OFF	ON	OFF	OFF
36 DECIMAL	24 HEX	OFF	OFF	ON	OFF	OFF	ON	OFF	OFF
35 DECIMAL	23 HEX	ON	ON	OFF	OFF	OFF	ON	OFF	OFF
34 DECIMAL	22 HEX	OFF	ON	OFF	OFF	OFF	ON	OFF	OFF
33 DECIMAL	21 HEX	ON	OFF	OFF	OFF	OFF	ON	OFF	OFF
32 DECIMAL	20 HEX	OFF	OFF	OFF	OFF	OFF	ON	OFF	OFF
31 DECIMAL	1F HEX	ON	ON	ON	ON	ON	OFF	OFF	OFF
30 DECIMAL	1E HEX	OFF	ON	ON	ON	ON	OFF	OFF	OFF
29 DECIMAL	1D HEX	ON	OFF	ON	ON	ON	OFF	OFF	OFF
28 DECIMAL	1C HEX	OFF	OFF	ON	ON	ON	OFF	OFF	OFF
27 DECIMAL	1B HEX	ON	ON	OFF	ON	ON	OFF	OFF	OFF
26 DECIMAL	1A HEX	OFF	ON	OFF	ON	ON	OFF	OFF	OFF
25 DECIMAL	19 HEX	ON	OFF	OFF	ON	ON	OFF	OFF	OFF
24 DECIMAL	18 HEX	OFF	OFF	OFF	ON	ON	OFF	OFF	OFF
23 DECIMAL	17 HEX	ON	ON	ON	OFF	ON	OFF	OFF	OFF
22 DECIMAL	16 HEX	OFF	ON	ON	OFF	ON	OFF	OFF	OFF
21 DECIMAL	15 HEX	ON	OFF	ON	OFF	ON	OFF	OFF	OFF
20 DECIMAL	14 HEX	OFF	OFF	ON	OFF	ON	OFF	OFF	OFF
19 DECIMAL	13 HEX	ON	ON	OFF	OFF	ON	OFF	OFF	OFF
18 DECIMAL	12 HEX	OFF	ON	OFF	OFF	ON	OFF	OFF	OFF
17 DECIMAL	11 HEX	ON	OFF	OFF	OFF	ON	OFF	OFF	OFF
16 DECIMAL	10 HEX	OFF	OFF	OFF	OFF	ON	OFF	OFF	OFF
15 DECIMAL	F HEX	ON	ON	ON	ON	OFF	OFF	OFF	OFF
14 DECIMAL	E HEX	OFF	ON	ON	ON	OFF	OFF	OFF	OFF
13 DECIMAL	D HEX	ON	OFF	ON	ON	OFF	OFF	OFF	OFF
12 DECIMAL	C HEX	OFF	OFF	ON	ON	OFF	OFF	OFF	OFF

Table 4-5. Switch 5: Setting the SmartNode Address (continued)

ADDRESS1		2	3	4	5	6	7	8		
11 DECIMAL	B HEX	ON	ON	OFF	ON	OFF	OFF	OFF	OFF	OFF
10 DECIMAL	A HEX	OFF	ON	OFF	ON	OFF	OFF	OFF	OFF	OFF
9 DECIMAL	9 HEX	ON	OFF	OFF	ON	OFF	OFF	OFF	OFF	OFF
8 DECIMAL	8 HEX	OFF	OFF	OFF	ON	OFF	OFF	OFF	OFF	OFF
7 DECIMAL	7 HEX	ON	ON	ON	OFF	OFF	OFF	OFF	OFF	OFF
6 DECIMAL	6 HEX	OFF	ON	ON	OFF	OFF	OFF	OFF	OFF	OFF
5 DECIMAL	5 HEX	ON	OFF	ON	OFF	OFF	OFF	OFF	OFF	OFF
4 DECIMAL	4 HEX	OFF	OFF	ON	OFF	OFF	OFF	OFF	OFF	OFF
3 DECIMAL	3 HEX	ON	ON	OFF						
2 DECIMAL	2 HEX	OFF	ON	OFF						
1 DECIMAL	1 HEX	ON	OFF							
* 0 DECIMAL	0 HEX	OFF								

* Addresses 0, 7E, and 7F, hex are reserved by the SmartNode protocol.

Table 4-6. Switch 7: Settings for RS-232 Port B

	Position			
	1	2	3	4
RI Input Port B				
232 RI connected	on			
forced active	off			
DTR Output Port B				
controlled by software	on			
forced active		off		
DCD Input Port B				
232 DCD connected			on	
forced active			off	
CTS Input Port B				
232 CTS connected				on
forced active				off

Table 4-7. Jumper W3: Setting DSR *

A-B	No connection
B-C	Forced active

* Setting Jumper W3 active (B-C) lets the SmartNode output DSR to the attached RS-232 device.

Appendix A: Sample Programs

This section contains two programs, one written in the GWBASIC programming language and one written in the C programming language, that are examples of how to program your PC to be the master device in your SmartNode network. These programs are by now means to be followed in every case. This sample supports four-wire duplex operation only. See Section 3.3 'Special Requirements' for more information on programming.

Program 1.

```

10  \SMART NODE DEMO PROGRAM
20  \PROGRAM WRITTEN IN GWBASIC
30  \
40  CLS                                \CLEAR THE SCREEN
50  OPEN "COM1: 9600,N,8,2,CS,DS,CD" AS #1  \OPEN COMMUNICATION PORT #1 AS
60                                     \FILE #1 USING 9600 BPS, 8
70                                     \DATA BITS, NO PARITY, 2 STOP
80                                     \DON'T LOOK FOR CS DS CD INPUTS
85                                     \RTS OUTPUT ACTIVE, RTS IS USED TO ACTIVATE THE
86                                     \RS-232 <-> RS-485 DRIVER ON THE RS-485 INTERFACE
87                                     \converter'
90  GOSUB 530                            \DISPLAY HELP SCREEN
100 NOTDONE = 1                          \INITIALIZE NOTDONE FLAG
110 \
120 \ MAIN PROGRAM LOOP
130 \ Data received in the 232 port (from network) is displayed on the screen reserved one byte commands."
140 \ Data from keyboard is transmitted out the 232 port except for a few."
150 \ These commands will allow the user to transmit the header string,"
160 \ trailer string, or reserved addresses out the "
170 \ 232 port by entering one byte control characters. "
180 \ See the HELP subroutine for additional details"
190 \
200 WHILE NOTDONE                        \LOOP UNTIL ^X ENTERED AT KEYBOARD
210 GOSUB 280                            \CHECK FOR KEYBOARD DATA
220 GOSUB 440                            \CHECK FOR 232 DATA
230 WEND
240 END                                  \CLOSE FILES AND EXIT PROGRAM
250 \
260 \SUBROUTINE TO CHECK FOR KEYBOARD DATA
270 \
280 A$=INKEY$: IF LEN(A$) = 0 THEN RETURN  \IF NO DATA FROM KEYBOARD: RETURN
290                                     \IF DATA FROM KEYBOARD IS:
300 IF ASC(A$) = 24 THEN NOTDONE=0       : RETURN    \^X = CHANGE FLAG TO EXIT PROGRAM
310 IF ASC(A$) = 20 THEN PRINT #1,"]!$@"; : RETURN    \^T = TX TRAILER OUT PORT
320 IF ASC(A$) = 8 THEN PRINT #1,"}&#/" ; : RETURN    \^N = TX HEADER OUT PORT
330 IF ASC(A$) = 4 THEN PRINT #1,CHR$(126); : RETURN   \^D = TX DESELECT ADDRESS
340 IF ASC(A$) = 2 THEN PRINT #1,CHR$(127); : RETURN   \^B = TX BROADCAST ADDRESS
350 IF ASC(A$) = 26 THEN PRINT #1,CHR$(0); : RETURN   \^Z = TX ADDRESS ZERO
360 IF A$ = "?" THEN GOSUB 530           : RETURN    \ ? = DISPLAY HELP SCREEN
370 IF ASC(A$) = 9 THEN GOSUB 810       : RETURN    \^I = DISPLAY SETUP INSTRUCTIONS
380 PRINT #1,A$;                        \IF NONE OF ABOVE, TX DATA FROM
390                                     \KEYBOARD OUT 232 PORT RS IS
400 RETURN
410 \
420 \SUBROUTINE TO CHECK FOR 232 DATA
430 \
440 WHILE NOT EOF(1)                    \LOOP UNTIL NO MORE 232 DATA
450     PRINT INPUT$(LOC(1),1);         \TO DISPLAY TO THE SCREEN
460 WEND
470 RETURN                              \THEN RETURN FROM SUBROUTINE
480 \

```

```

490  `SUBROUTINE TO TRANSMIT HELP SCREEN
500  `
510  PRINT ""
520  PRINT ""
530  PRINT ""
540  PRINT ""
550  PRINT "          SMARTNODE NETWORK TEST PROGRAM"
560  PRINT " . Data received in the 232 port (from the network) is displayed on the screen"
570  PRINT " . Data from keyboard is transmitted out the 232 port except for:"
580  PRINT "   `^B - tx the broadcast address (127)   ^D - tx the deselect address (126)";
590  PRINT "   `^H - tx the header '}&#/'           ^I - display setup instructions"
600  PRINT "   `^T - tx the trailer '!$@'           ^X - exit this program"
610  PRINT "   `^Z - tx the master address (000)     ? - display this
        help screen"
620  PRINT " NOTE: to enter ^X depress the control key at the same
        time as the X key"
630  PRINT ""
640  PRINT " OPERATION:          YOU ENTER:"
650  PRINT " Broadcast data to all nodes      ^H ^B   <data>"   ^T
660  PRINT " Just tx data to a node           ^H <address> <data>" ^H^D
670  PRINT " Give node permission to Tx (We Rx) ^H <address>      ^T"
680  PRINT " Full-Duplex Tx                   ^H <address> <data>"
690  PRINT "           then Rx                                     ^T"
700  PRINT " Full Duplex Rx                   ^H <address>      ^T"
710  PRINT "           and Tx (simultaneously) ^H <address> <data>" ^T"
720  PRINT " Stop Rx (Full Duplex/4-wire)      ^H ^D"
740  PRINT ""
750  PRINT " NOTE Use the keyboard to enter the <address> and <data>
        fields"
760  PRINT " For example, the A-key will transmit address 065 (see
        ASCII chart)"
770  RETURN
780  `
790  `SUBROUTINE TO TRANSMIT SETUP INSTRUCTIONS
800  `
810  PRINT ""
820  PRINT ""
830  PRINT "I SMARTNODE LOOPBACK TESTING SETUP INSTRUCTIONS:"
840  PRINT " . Connect 232 port of SmartNode to COM1 port of PC with a cross cable"
850  PRINT " . Configure SmartNode for 9600 baud, 8 data, 2 stop, XON/XOFF,"
860  PRINT "   Disable both timeouts, set Node address to 000"
870  PRINT " . Loop the transmit pair to the receive pair on the 485 port"
880  PRINT " . transmit address zero to attached SmartNode (with no preceding header)"
890  PRINT "   Data from keyboard will be sent out port, SmartNode should"
900  PRINT "   loop this data back to the PC (which would be displayed on screen)"
905  Print "   If problems, reset unit and try again"
910  PRINT ""
920  PRINT "II SMARTNODE NETWORK TESTING SETUP INSTRUCTIONS"
930  PRINT " . Connect 232 port of a 232-485 interface converter to COM1 port of PC"
940  PRINT "   Configure converter's 232 port as DCE (or use cross over cable)
950  PRINT " . Attach one or more SmartNodes to the 485 port of the interface converter"
960  PRINT "   Use a 4-wire network"
970  PRINT " . Configure SmartNode's 485 ports for 9600 baud, 8 data, 2 stop"
980  PRINT " . attach your 232 equipment to the SmartNode(s)"
990  PRINT " . Configure each SmartNode's 232 ports to match the attached 232 equipment"
1000 PRINT " . Set each SmartNode for a unique address (except 000, 126, and 127)"
1010 PRINT " . Configure the other SmartNode options to match your preference"
1030 PRINT " . Follow the instructions in the help menu ('?') to test the network"
1040 PRINT "   (ie, broadcast data to nodes, tx & rx data from a particular node)"

```

1050 RETURN

Program 2: Turbo C

This sample supports four-wire duplex operation only. See Section 3.3 'Special Requirements' for more information.

```

/* SmartNode Demo program written to run under Turbo C */

#include "stdio.h"
#include "stdlib.h"
#include "bios.h"

#define ok_to_tx                0x2000
#define rx_char_avail          0x100
#define return_port_status     3
#define rx_a_char              2
#define tx_a_char              1
#define initialize_port        0
#define to_9600bps_noparity_2stop_8data 0xe3
#define to_9600bps_noparity_2stop_7data 0xe2
#define to_1200bps_noparity_2stop_8data 0x83
#define to_1200bps_noparity_2stop_7data 0x82
#define enterkey               0x0d
#define com1                   0
#define bell                    0x07

char input,data;
int serial_port_status,new_address,done=0;

main()
{

int done = 0;                                /*local variable for main function */

printf("\n\nTo execute all options (except #6) you need to have a 4-wire Smart Node");
printf("\n\nnetwork attached to the COM1 port of your PC. A 232-485 interface");
printf("\n\nconverter should be used to connect the PC to the network. Configure");
printf("\n\nthe 485 port of all Smart Nodes for 9600 bps, 8 data bits, no parity.");
printf("\n\nNOTE: Make sure CTS, DSR, RTS, and DCD 232 are active at COM1 port!");

/* initialize communication port to 9600 bps, no parity, 2 stop, and 8 data bits */

bioscom(initialize_port, to_9600_noparity_2stop_8data, com1);

/* display options menu and prompt user for his/her choice */

while (!done)
{
printf("\n\nMENU\n");
printf("\n(1) Deselect all nodes (remove permission to transmit)");
printf("\n(2) Broadcast test message to all nodes");
printf("\n(3) Transmit test message to a specific node");
printf("\n(4) Give a specific node permission to transmit");
printf("\n(5) Establish a Full-Duplex link with a specific node");
printf("\n(6) Run loopback test with a Smart Node attached to COM1 port");
printf("\n(7) Exit");
printf("\n\nEnter your choice: ");

```

```

while ((input=getch()) < '1' || input > '7') putchar(bell);
switch(input)
{
    case '1':
        deselect();
        break;
    case '2':
        broadcast();
        break;
    case '3':
        transmit_only();
        break;
    case '4':
        receive_only();
        break;
    case '5':
        full_duplex_link();
        break;
    case '6':
        loopback();
        break;
    case '7':
        done=1;
        break;
}
}

deselect()
{
    tx_header();
    tx_deselect_address();

    printf("\n\nPermission to transmit removed from all Smart Nodes");
}

broadcast()
{
    tx_header();
    tx_broadcast_address();
    tx_test_data();
    tx_trailer();

    printf("\n\nTest data transmitted to all nodes");
}

transmit_only()
{
    get_address();

    tx_header();
    tx_address(new_address);
    tx_test_data();
}

```

```
    tx_header();                                /*Do not give node */
    tx_deselect_address();                       /* permission to transmit*/
    printf("\n\nTest data transmitted");
}
receive_only()
{
/* clear 232 receive buffer of any characters and get Smart Node address from user */

    clear_receiver();
    get_address();

    tx_header();
    tx_address(new_address);
    tx_trailer();

/* NOTE: the Node is allowed to transmit when it receives a valid trailer string
   following a valid header string and it's unique address (data optional) */

    printf("\n\nNode given permission to transmit, data received will be displayed to screen.");
    printf("\nEnter any key to deselect node and exit!\n\n");
    done=0;

/* loop displaying data received in 232 port to the screen until the user enters any key */

    while(!done)
        {
            if (kbhit() !=0)                    /* was keyboard hit? */
                {
                    data=getch();              /* if so - prepare to exit */
                    done = 1;
                }
            port_to_screen();                    /*display 232 port data to screen */
        }

/*before exit- remove permission to transmit from node by sending the deslect address */

    tx_header();
    tx_deselect_address();
}

full_duplex_link()
{
    printf("\n\nA full duplex link will be established with the node of your choice.");
    printf("\nEnter ^X to exit!\n\n");
}
```

```

/*clear 232 receive buffer of any data and prompt user for Smart Node address*/

clear_receiver();
get_address();

tx_header();
tx_address(new_address);
tx_trailer();           /*node now given permission to transmit */

tx_header();           /*address node again to we can */
tx_address(new_address); /*transmit to it while we are also receiving */

terminal_mode();      /*data in port to screen and data from keyboard out port */
}

terminal_mode()

tx_header ();          /*Remove nodes permission to transmit*/
tx_deselect address_address ();

{
done=0;

/*continue to stay in terminal mode until a ^X is entered */

while (!done)
{
port_to_screen();
keyboard_to_port();
}
}

port_to_screen()
{
if (check_rx_char_in_yet() !=0)          /*any 232 data received yet? */
{
input=get_rxed_char();          /*if so - remove it from buffer */
putchar (input);              /*and display it to screen */
}
}

keyboard_to_port()
{
if (kbhit() !=0)                /*any keyboard data available?*/
{
data=getch():                  /*if so- get data */

switch(data)
{
case 0x18:                      /* if ^X - prepare to exit */
done=1;
break;

default:
tx_byte_out_port(data); /*if anything else- transmit it out port*/
break;
}
}
}
}

```

```

loopback()
{
    printf("\n\nAttach smart node directly to the COM1 port of your PC.");
    printf("\nConfigure the unit as described in the user's manual under");
    printf("\nthe section Loop-Back testing.");
    printf("\nNOTE: Make sure CTS, DSR, and DCD 232 inputs are active at COM1 port!");
    printf("\n\nIf unit configured correctly and it is operational you");
    printf("\nshould receive an echo of all data you enter at the keyboard!");
    printf("\n(Reset the unit before trying this option again if you experience problems)");
    printf("\n\nNOTE: enter ^X to exit!\n\n");
    clear_receiver();

    tx_host_address();                /* first thing we tx is address zero */
    terminal_mode();                 /* note: no preceding header string */
    tx_trailer();
}

/* empty 232 port buffer of all characters */

clear_receiver()
{
    while (check_rx_char_in_yet() != 0) get_rxed_char();
}

tx_broadcast_address()
{
    new_address='\x7F';              /* broadcast address = 7F hex or 127 decimal */
    tx_address(new_address);
}

tx_deselect_address()
{
    new_address='\x7E';              /* deselect address = 7E hex or 126 decimal */
    tx_address(new_address);
}

tx_host_address()
{
    new_address='\x00';              /* host/master address = zero */
    tx_address(new_address);
}

/* get_address function prompts the user for a 3 digit decimal address followed by the <enter> key */

get_address()
{
    printf("\n\nEnter Address of a specific Smart Node in decimal (ie, 065): ");
    scanf("%d",&new_address);
}

/* transmit the Smart Node address passed to this function out the 232 port */

tx_address(passed_address)
int passed_address;
{
    tx_byte_out_port(passed_address);
}

```

```

}
/* transmit the four byte header string out the 232 port */

tx_header()
{
    tx_string_out_port("}&#/"");
}

/* transmit the four-byte trailer string out the 232 port */

tx_trailer()
{
    tx_string_out_port("]!$@"");
}

/* transmit test data out the 232 port */

tx_test_data()
{
    tx_string_out_port("\n\nThis is a test data sent from the PC.\n\n");
}

/* transmit a string (a byte at a time) out the 232 port whose address was passed to this function */

tx_string_out_port(string_pntr)
char*string_pntr;
{
    char nextbyte;
    nextbyte=*string_pntr++;          /*tx a byte at a time until at end of string */
    while (nextbyte != NULL)
    {
        tx_byte_out_port(nextbyte);  /*tx next byte out 232 port */
        nextbyte=*string_pntr++;     /* get next byte of string */
    }
}

/* transmit the byte passed to this function out the 232 port */

tx_byte_out_port(byte)
char    byte;
{
    while (check_txer_ready() == 0);  /*wait until the 232 serial chip is ready to transmit */
    tx_char(byte);                    /*then transmit the byte */
}

```

```
/* poll the 232 serial chip for status and return only the status of the transmitter */

check_txer_ready()
{
    serial_port_status = bioscom(return_port_status,0,com1);    /* poll chip */
    return(serial_port_status & ok_to_tx);                    /* return status of tx'er */
}

/* tell the 232 serial chip to transmit this byte out the COM1 port */

tx_char(byte)
char byte;
{
    bioscom(tx_a_char,byte,com1);
}

/* poll the 232 serial chip for status and return only the status of the receiver */

check_rx_char_in_yet()
{
    serial_port_status = bioscom(return_port_status,0,com1);    /* poll for status */
    return (serial_port_status & rx_char_avail);                /*return status of rx'er */
}

/* tell the 232 serial chip to return the next byte available in its receive buffer */

get_rxed_char()
{
    return(bioscom(rx_a_char,0,com1));
}
```

Appendix B: RS-232 and RS-485 Pinning

The table below shows the pinning for the RS-232 connector on the SmartNode.

DB9 Pin (female)	Name	Description	Source	Direction to SmartNode *
3	TD	Transmit Data	DTE	(output)
2	RD	Receive Data	DCE	(input)
7	RTS	Request to Send	DTE	(output)
8	CTS	Clear to Send	DCE	(input)
6	DSR	Data Set Ready	DCE	(input)
5	SG	Signal Ground	DTE and DCE	
1	DCD	Data Carrier Ready	DCE	(input)
4	DTR	Data Terminal Ready	DTE	(output)
9	RI	Ring Indicator	DCE	input)

Table B-1. RS-232 Connector Pinning

* The SmartNode RS-232 port is factory set to DTE. It can operate as DCE only with a cross over cable, in which case the direction of the signals is reversed.

Table B-2. RS-485 Terminal Block Layout

	Screws			
	1	2	3	4
Signals	RDB	RDA	TDB	TDA

Appendix C: TESTING THE UNITS

The SmartNode units can be configured to run a simple RS-485 loopback test. Data can be received in the RS-232 side, looped at the RS-485 side, and transmitted back to your RS-232 device. Configure the unit for the test as follows:

- Configuring the SmartNode:
 - set the address to zero (set all positions of DIP switch SW5 to OFF)
 - match the baud rate, word structure, and flow control of your RS-232 device (i.e., a nonaddressable terminal at 9600 baud, 8 data, 1 stop, XON/XOFF)
 - configure the RS-485 baud rate and word structure to any setting except for fewer data bits than you are using on the RS-232 port
 - disable both the SESSION and NO TRANSMIT timeouts
 - always buffer RS-232 data
 - set the RS-232 jumper options to match the flow control of your attached RS-232 device
- loop the transmit pair to the receive pair on the RS-485 interface (TDA to RDA and TDB to RDB)
- connect your RS-232 test device (use a cross over cable if it's DTE)

Note: To start the loop back test you must first transmit a 00 hex (^@) to the RS-232 port of the SmartNode. All other additional data received should be echoed back to your RS-232 device.

If the loopback test does not work, reset the SmartNode and try again. If, after several tries the test does not work, contact your technical support representative.



© Copyright 1994. Black Box Corporation. All rights reserved.

1000 Park Drive • Lawrence, PA 15055-1018 • 724-746-5500 • Fax 724-0746-0746