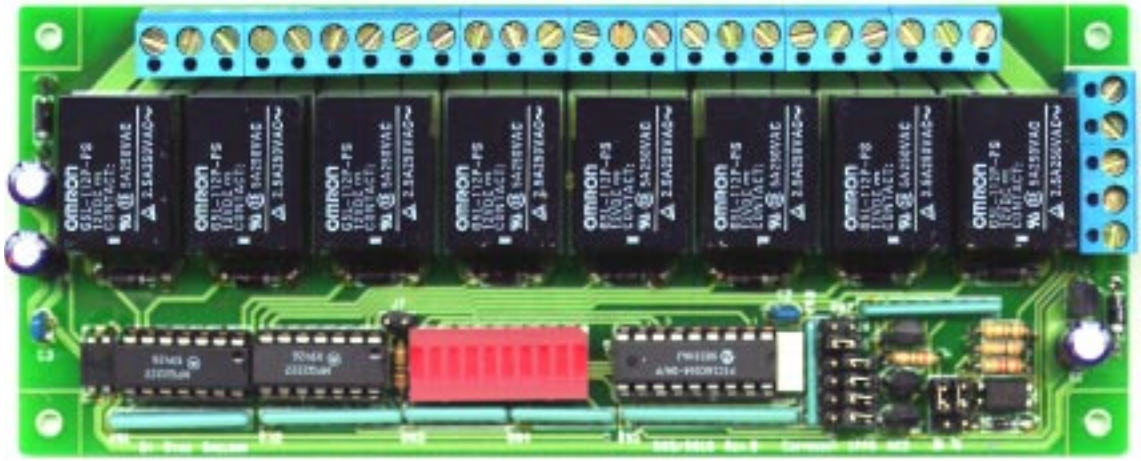
R85/R810 RS-232 Relay Driver



Revision B

NCD

Users Guide

Revision 0.9 (unedited)

By Ryan Sheldon

Disclaimer:

NCD will not be responsible for any personal injury or damage to computers or other electronic devices caused in conjunction with the use of NCD products. The user assumes full responsibility of any and all damage that may result in combination with NCD products, services, and literature. NCD takes EVERY known precaution to ensure documentation and examples are accurate. Return this device within 30 days of purchase for a full refund if you do not agree with these terms and conditions. Operation or assembly of any products sold by NCD signifies your agreement with these terms.

Special Thanks:

I would like to thank Kima Dirks for the many hours she has spent editing and refining all of my written literature. I would also like to thank Tom Maier for his dedication to this project. Without his help our products would not be possible.--Ryan Sheldon

Introduction and Overview:

Thank you for choosing the NCD-R8x Revision B relay driver. The R8x series is perhaps the smallest feature packed relay driver on the market.

Up to 16 R8x series controllers will coexist on a single serial line, allowing 128 relays to be fully controlled from the serial port of your computer. Combine with our LCD driver to display the status of the relays. Combine with our other products to enable real-world control from your computer.

We developed a simple networking system to allow crossplatform control of our products. In this manual, we will discuss how our proprietary networking system works and how it can be used in real-world applications.

We have integrated many features directly on the board simply not found in any other controller on the market. From a hearbeat LED that tells you everything is running to 2-way communication at speeds up to 19.2K baud, the R8x offers more "bang for the buck" than any other controller made by any other compeditor. We hope you enjoy building your relay driver and hope you will gain the knowledge of how to control the outside world in the process.

What You Will Need:

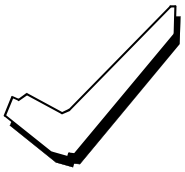
To use the R8x series, you will need the following items: Computer with 3.5" drive (PC, or Macintosh), 12 volt DC power supply, wire, and serial cable (discussed on the next page). A wrist strap should be used for static grounding during installation of all semiconductors.

Helpful hint.

Could cause problems later under normal operation.

Fatal mistake.

The X tells about common assembly and operation errors as well as other things to watch out for.



The check is a guide to help the user with assembly and operation. Use them to prevent mistakes.

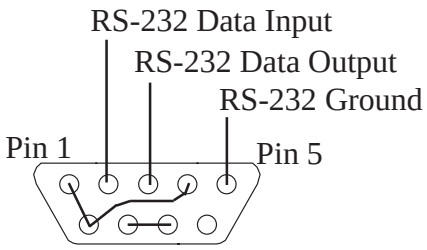
Interfacing to the Serial Port of Your Computer

Before we begin, we need to build a cable to transmit data from the serial port of your computer. Since most problems can be attributed to improper connection to the serial port, extra time should be spent verifying these simple connections. Follow the diagrams to build a cable for the system you intend to use. Take care not to accidentally short out pins on the connectors. Though it is theoretically OK to short all the lines together from an RS-232 port, we strongly recommend that you take every precaution to avoid this. Once the cable is built, connect an LED with a 2.2K resistor to the RS-232 ground and RS-232 data output lines. The ground should connect to the short lead of the LED. The resistor should connect to the long lead. The remaining end of the resistor should connect to the RS-232 data line. Enter and run the program for your computer found to the right of each box. You should see the LED flash so fast that it appears to be on. Do not proceed until your cable is working. *Please note that all illustrations show the **solder side** of the connector.*

PC Specific:
If you are using a PC, identify the type of serial connector you will need. There are only two common choices available. Both connectors are female; the first diagram is for a DB-25 connection, and the second is for a DB-9 connection. PC users will need to find out if the serial port is accessed as "COM1:" or "COM2:". The best way to do this is to try both software settings and see which one works. Software from here on will assume you are using "COM2:".

Macintosh Specific:
If you are using a Macintosh system, make sure the connections are not accidentally touching each other. The lines on the mini 8-pin din are very close together and very difficult to work with.

PC Users

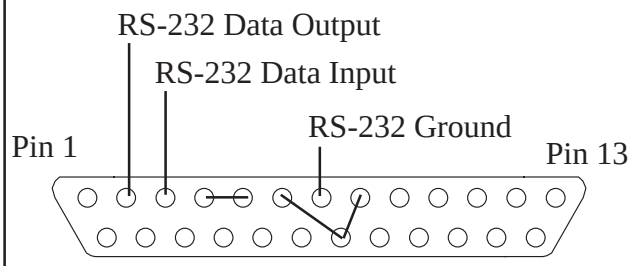


Example Shown Using a DB-9 Female Connector

In QBasic for PC compatible systems, enter and run the following program to test the serial port of your computer:

```
OPEN "com2:9600,N,8,1,CD0,CS0,DS0,OP0" FOR OUTPUT AS #1
DO
    FOR N = 0 TO 255
       PRINT #1, CHR$(N);
    NEXT N
LOOP
```

PC & Amiga Users

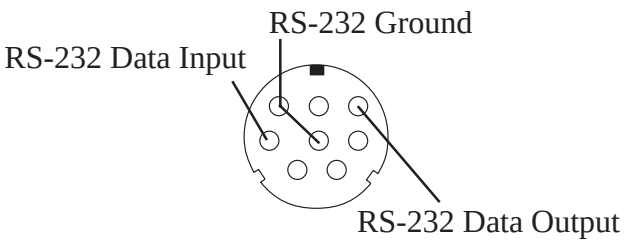


Example Shown Using a DB-25 Female Connector

In Amos for Commodore Amiga systems, enter and run the following program to test the serial port of your computer. Press the mouse button to exit the program.

```
Serial Open 0,0
Serial Speed 0,9600
Serial Buf 0,64
Serial Bits 0,8,1
While Mousekey=0
    For N = 0 To 255
        Serial Send 0,Chr$(N)
    Next N
Wend
```

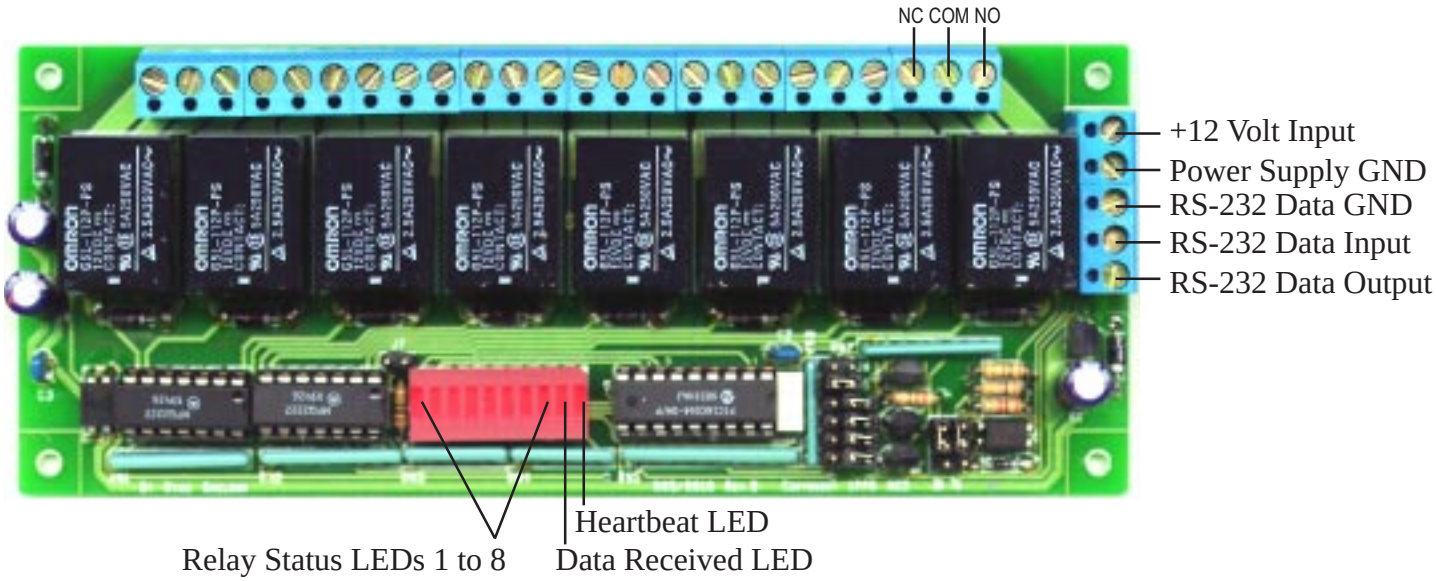
Macintosh Users



Example Shown Using a MINI DIN8 Female Connector

In Chipmunk Basic for Apple Macintosh systems, enter and run the following program to test the serial port of your computer. Be sure to select the serial port your cable is connected to (modem or printer). Press the mouse button to exit the program.

```
10 open "COM1:" for output as #1
20 while not mouse(0)
30    for n = 0 to 255
40       fputbyte n,1
50    next n
60 wend
```



Jumper	Default Setting	Description
J1	Removed	Address Select
J2	Removed	Address Select
J3	Removed	Address Select
J4	Removed	Address Select
J5	Installed	Baud Select 1
J6	Removed	Baud Select 2
J7	Installed	Data LED Brightness
J8	Installed Between Upper Two Posts	Data Output Type
J9	Installed Between Upper Two Posts	Data Input Type

The table shown at left gives a very brief description of the function of the jumpers on the R8x controller. When troubleshooting the R8x, set the jumpers to the default setting as shown in the photo and in the table.

Relay Connections

NO: Normally Open - This connection is not connected to the common line of the relay unless the relay is switched on.

COM: Common - This connection switches between NC and NO depending on if the relay is turned off or on.

NC: Normally Closed - When a relay is in its off state, the COM connection is connected to this terminal.

Baud	J5	J6
1200	Removed	Removed
9600	Installed	Removed
19200	Removed	Installed
Reserved	Installed	Installed

The table above shows how to set the jumpers for 1 of 3 possible baud rates. Note that J5 and J6 should NOT have jumpers installed at the same time. The R8x Revision B only reads the address and baud rate jumpers when first powered. To change the device number or baud rate, jumpers must be changed and power must be removed and reapplied for changes to take effect.

Special Function Jumpers	Function Description
J7 Data Input LED Brightness Control	Install for 9600 or 19.2K Baud operation, remove for 1200 Baud.
J8 Installed Between Upper Two Posts	Sets Data Output to Logic Level RS-232 Signal
J8 Installed Between Lower Two Posts	Sets Data Ouptut to Open Collector RS-232 Signal
J9 Installed Between Upper Two Posts	Sets RS-232 Data Input for PC Systems (Standard RS-232)
J9 Installed Between Lower Two Posts	Sets RS-232 Data Input for Macintosh Systems (RS-422)

NOTE: J8 and J9 must be set for proper communication. If you are using the RSB serial booster as an interface to your PC, you must set the data output of the R8x for Open Collector communications (J8 installed between lower two posts). If you are controlling a single R8x and are not using an RSB or any other NCD devices, you can set the output to Logic Level RS-232 communications. Note that the output is NOT true RS-232, but is compatible with most PCs, but is NOT compatible with Macintosh systems. Note that the RS-232 Data Output wire is optional. J8 may be set in either position of the Data Output wire is not connected to your computer.

Configuring and Controlling the R8x Relay Driver

NCD Device Number	ASCII Command Range	J1	J2	J3	J4
0	0 to 15	Removed	Removed	Removed	Removed
1	16 to 31	Installed	Removed	Removed	Removed
2	32 to 47	Removed	Installed	Removed	Removed
3	48 to 63	Installed	Installed	Removed	Removed
4	64 to 79	Removed	Removed	Installed	Removed
5	80 to 95	Installed	Removed	Installed	Removed
6	96 to 111	Removed	Installed	Installed	Removed
7	112 to 127	Installed	Installed	Installed	Removed
8	128 to 143	Removed	Removed	Removed	Installed
9	144 to 159	Installed	Removed	Removed	Installed
10	160 to 175	Removed	Installed	Removed	Installed
11	176 to 191	Installed	Installed	Removed	Installed
12	192 to 207	Removed	Removed	Installed	Installed
13	208 to 223	Installed	Removed	Installed	Installed
14	224 to 239	Removed	Installed	Installed	Installed
15	240 to 255	Installed	Installed	Installed	Installed

Device Addressing

The R8x has a set of 4 address jumpers (J1-J4) used to set the address of the relay driver on the NCD RS-232 network. A software address is a number from 0 to 15 used to speak to different NCD devices. Every NCD device has a set of address jumpers that set its device number. Device 0 may be a relay driver board while device 1 may be a video switcher. Device numbers provide a convenient way of sending commands to multiple devices connected to a single serial port.

A serial port can send up to 256 ASCII characters ranging from 0 to 255. When an NCD device is set to address 0, the device will only listen to ASCII characters 0 to 15. When a device is set to address 1, it will only listen to ASCII characters 16 to 31. So the address jumpers determine which set of ASCII characters a device will listen to.

NOTE: The R8x Revision B only reads the address and baud rate jumpers when first powered. To change the device number or baud rate, jumpers must be changed and power must be removed and reapplied for changes to take effect.

Sending Commands to the R8x Relay Driver

The R8x relay controller listens to 18 different commands for controlling and communicating the status of the relays with the host computer. When set to device 0, ASCII characters 0 to 7 are used to turn off relays 1 to 8. ASCII characters 8 to 15 are used to turn on relays 1 to 8 accordingly. Sending ASCII character 7 three consecutive times will turn off relay 8 and reports the status of the 8 relays. Similarly, sending ASCII character 15 three times will turn on relay 8 report the status of the 8 relays.

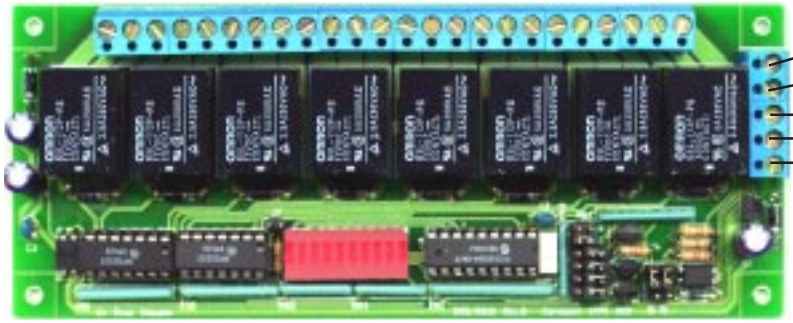
ASCII characters 16 to 31 can be used to speak to an R8x set to address 1 by installing jumper J1 (see table above). ASCII characters 16 to 23 will turn off relays 1 to 8 accordingly. ASCII characters 24 to 31 will turn on relays 1 to 8. Sending ASCII character 23 three times consecutively will turn off relay 8 and report the status of all 8 relays. Sending ASCII character 31 three times consecutively will turn on relay 8 and report the status of all 8 relays.

When a command is sent asking the R8x relay driver for the status of the 8 relays, a number from 0 to 255 will be returned to the computer. The following program algorithm shows how to decode this number and determine if each relay is on:

```
RN= Number Returned from the R8x Relay Driver
IF RN>=128 then Relay 8 is On, So Subtract 128 from RN
IF RN>=64 then Relay 7 is On, So Subtract 64 from RN
IF RN>=32 then Relay 6 is On, So Subtract 32 from RN
IF RN>=16 then Relay 5 is On, So Subtract 16 from RN
IF RN>=8 then Relay 4 is On, So Subtract 8 from RN
IF RN>=4 then Relay 3 is On, So Subtract 4 from RN
IF RN>=2 then Relay 2 is On, So Subtract 2 from RN
IF RN=1 then Relay 1 is On, So Subtract 1 from RN
RN Should Now equal 0.
```

Actual programming examples will be provided later in this manual, but at least you have an idea of how the algorithm works. Displaying the binary value of the returned number will also indicate which relays are active. A 1 would indicate the on state while a 0 would indicate the off state.

ASCII Character	Function
0	Turn Off Relay 1
1	Turn Off Relay 2
2	Turn Off Relay 3
3	Turn Off Relay 4
4	Turn Off Relay 5
5	Turn Off Relay 6
6	Turn Off Relay 7
7	Turn Off Relay 8
7, 7, 7	ASCII 7 Sent 3 Times Turn Off Relay 8 and Report the Status of all 8 Relays.
8	Turn On Relay 1
9	Turn On Relay 2
10	Turn On Relay 3
11	Turn On Relay 4
12	Turn On Relay 5
13	Turn On Relay 6
14	Turn On Relay 7
15	Turn On Relay 8
15, 15, 15	ASCII 15 Sent 3 Times Turn On Relay 8 and Report the Status of all 8 Relays.



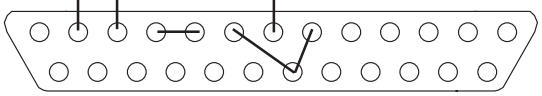
+12 Volt Power Supply Input

Power SupplyGround

Short Power Supply Ground and Data Ground only if Data will be sent back to the computer.

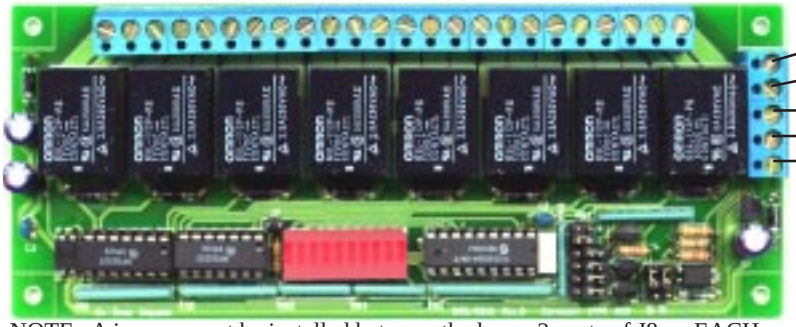
Pin 1

Pin 13



NOTE: A jumper must be installed between the upper 2 posts of J8 for the R8x to reply back to the computer when wired to your PC as shown in this diagram. The connection to Pin 3 of the DB-25 is optional, and is only used to report the status of the 8 relays.

Example: DB-25 Female Connector, Solder Side Shown.



+12 Volt Power Supply Input

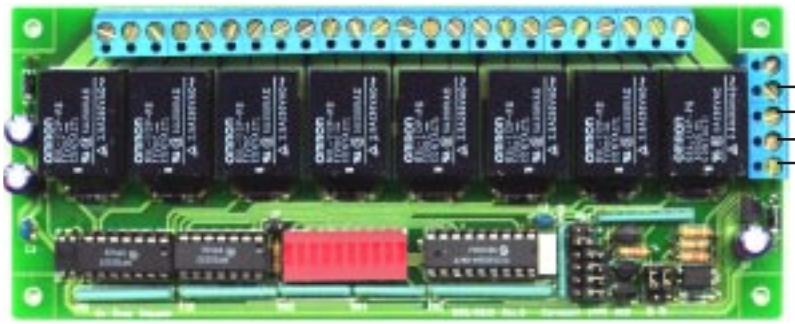
Power SupplyGround

Short Power Supply Ground and Data Ground only if Data will be sent back to the computer.

Optional Wire: This wire is used to send the status of the 8 relays back to the computer upon software request. This wire is not required and may be omitted for "blind" control applications.

NOTE: A jumper must be installed between the lower 2 posts of J8 on EACH R8x controller to set the RS-232 data output to Open Collector transmission. Power must be applied to each relay controller. Each controller should be set to a different device number using jumpers J1 through J4.

NOTE: When set to device 1, the R8x will respond to ASCII characters 16 to 31. ASCII Characters 16 to 23 will turn off each of the 8 relays while characters 24 to 31 will turn them on. Sending ASCII character 23 three times will turn off relay 8 and report the status of the 8 relays. Sending ASCII character 31 three times will turn on relay 8 and report the status of the 8 relays.



NOTE: When set to device 2, the R8x will respond to ASCII characters 32 to 47. ASCII Characters 32 to 39 will turn off each of the 8 relays while characters 40 to 47 will turn them on. Sending ASCII character 39 three times will turn off relay 8 and report the status of the 8 relays. Sending ASCII character 47 three times will turn on relay 8 and report the status of the 8 relays.

The RSB serial booster must be powered by 15 to 24 volts DC. A 12 Volt DC wall adapter usually provides 15 volts, which works well for powering the RSB serial booster. If you have communications problems, check the input voltage of the RSB serial booster.

Initial Test:

At this point, we will assume you followed our directions for building and testing a serial cable to connect between your computer and relay controller. We recommend driving this board at 12 volts. A suitable power supply should now be connected. Take care NOT to connect the power supply backwards. See the diagram on the next page for complete details for connecting the relay driver.

Before using the R85/R810, remove address jumpers J1 to J4. Install jumpers J5 and J7. J6 should NOT be installed. J8 and J9 should be installed for your type of connection and computer system. When power is applied, the heartbeat light should begin flashing rapidly indicating the on-board microprocessor is operating correctly. Verify software control by running the program "RELAY8.BAS." Modification may be required to this program depending on the system you are using. If you performed the LED test when you were building your cable, you may have found that you need to use "COM1:" instead of "COM2:". In this case, make the appropriate programming change before proceeding. The test program was written to turn each relay on, then off. When data is received, the "Data Received LED" should be flashing. The program "RELAY8B.BAS" should be used to report the status of the 8 relays back to the computer.

Debugging:

Basic checks: Verify all chips are located and oriented in the proper location. Also check make sure all pins are in the socket. Verify all parts are installed according to the directions and that your serial cable is functioning properly by performing the LED test on the third page. Also verify that there are no shorted solder connections on the board. We will also assume the board is oriented in the direction shown on the front cover.

Problem: Heartbeat LED Does Not Flash.

Try: Make sure CR1 is correctly soldered in.
Make sure DC power supply is at least 10 volts, voltage may drop when attached, veryify voltage has not dropped below 10 volts.
Make sure U6 and U4 are properly installed.
U6 resembles a transistor, verify that it is a 78L05 voltage regulator.
Verify D9 and D11 are properly installed.
Verify +5 volts on bottom left pin of U1.
Make sure U1 and LEDA are properly installed. LEDA should have lettering facing the relays.

Problem: Data LED Does Not Flash.

Try: Make sure J8 is installed between upper two posts for PC systems and lower two posts for Macintosh systems.
Install J7 to see if the LED is unusually dim.

Problem: Relay LEDs Do Not Light, but Data is Received and Heartbeat Flashes.

Try: Make sure the device number in software matches the device number in hardware (Jumpers J1 to J4).
LEDA, RN3, or RN4 may be installed backwards. Verify orientation.

Problem: Relay LEDs Light but Relays (some or all) Do Not Actuate.

Try: U2 and U3 cannot be installed backwards, they are not orientation specific. One or more transistors in the U2 or U3 transistor arrays may be bad.
Make sure RN1 and RN2 are properly installed.

Problem: Relays Actuate, but Status Cannot be Read.

Try: RS-232 Data Ground and Power Supply Ground must be tied together to report data back to the computer.
J8 controls the data output type. While the data output does not conform to RS-232 standards, most PCs accept data at logic levels (0 to 5 volts).
Make sure J8 is installed between the upper two terminals for RS-232 communications. This mode does NOT allow more than 1 R8x to be attached to a single RS-232 serial port, but does not require you to use an NCD-RSB serial booster. If a jumper is installed at J8 between the lower two terminals, an NCD-RSB serial booster will be required. This mode sets the data output for Open Collector (O.C.) RS-232 communications, and is ONLY compatible with the NCD-RSB serial booster. This mode must be used when attaching multiple relay drivers to a single serial port. The NCD-TR32 is required for communication with a Macintosh system.

Special Notes About the R8 Series Relay Drivers:

The R85 and R810 revision B is a complete redesign of the popular NCD-R85/R810 relay driver. Many new features have been added to the design making it the smallest, most powerful relay driver on the market. While infrared communications is no-longer an on-board option, an external infrared receiver may still be added. New Revision B features include:

FET enabler, firmware bootup, and native logic pull states provide triple protection to keep relays from "clicking" on powerup.
29% reduction in physical board size accomplished by software emulation of a 4099 IC found on Revision A boards.
Supports 19.2K baud operation with future hardware support for 38.4K baud with firmware upgrade.
Reports status of all 8 relays with 2 new commands while still 100% software compatible with Revision A software.
Reduced parts count and in-line design for quick assembly and easy trouble-shooting.
Jumper selectable open collector and RS-232 data outputs.
Clamping capacitor/diode system absorbs dangerous flyback voltages before they are allowed to reach driver/control circuits.
Dual power distribution system keeps polarity protection components cool for long-term operation.
Overkill design uses all components at a fraction of their rated values to extend product longevity.

Programming Examples:

```
OPEN “COM2:9600,N,8,1,CD0,CS0,DS0,OP0” FOR RANDOM AS #1
Device = 0
DO
  ‘Clear Output Screen
  CLS
  PRINT “Randomly Setting The Status of 8 Relays”
  ‘Randomly Set all 8 Relays
  FOR Relay = 1 TO 8
    RN = INT(RND(1) * 2): REM Set Random Value of Relay
    RELAY8 Device, Relay, RN
    ‘Ask Relay Driver for Current Status By Sending Random Value
    ‘of Relay 8 two additional Times
    IF Relay = 8 THEN
      RELAY8 Device, Relay, RN
      RELAY8 Device, Relay, RN
    END IF
  NEXT Relay
  PRINT “Waiting for Reply”
  IN = ASC(INPUT$(1, 1)): REM Read the data that returned from the R8x
  PRINT
  PRINT “Returned:”; IN: REM Display Returned Data
  PRINT
  PRINT “All Relays are Off Except for the Following:”
  ‘Test to see of Relay 1 is On by Anding it with Returned Value
  IF (1 AND IN) = 1 THEN PRINT “Relay 1 is On”
  ‘Test to see of Relay 2 is On by Anding it with Returned Value
  IF (2 AND IN) = 2 THEN PRINT “Relay 2 is On”
  ‘Test to see of Relay 3 is On by Anding it with Returned Value
  IF (4 AND IN) = 4 THEN PRINT “Relay 3 is On”
  ‘Test to see of Relay 4 is On by Anding it with Returned Value
  IF (8 AND IN) = 8 THEN PRINT “Relay 4 is On”
  ‘Test to see of Relay 5 is On by Anding it with Returned Value
  IF (16 AND IN) = 16 THEN PRINT “Relay 5 is On”
  ‘Test to see of Relay 6 is On by Anding it with Returned Value
  IF (32 AND IN) = 32 THEN PRINT “Relay 6 is On”
  ‘Test to see of Relay 7 is On by Anding it with Returned Value
  IF (64 AND IN) = 64 THEN PRINT “Relay 7 is On”
  ‘Test to see of Relay 8 is On by Anding it with Returned Value
  IF (128 AND IN) = 128 THEN PRINT “Relay 8 is On”
  PRINT
  PRINT “Press Any Key.”
  WaitKey a$
LOOP

SUB RELAY8 (Device!, Relay!, Status!)
  ‘Send computed data to Relay8 device.
  PRINT #1, CHR$((Device * 16) + (Relay - 1 + (8 * Status)));
END SUB
```

The program at left can be found on the included disk under the file name "RELAY8B.BAS". This QBasic source code shows how easy it is to control the relay driver. The Open statement on the first line opens the serial port for RS-232 communications. The subroutine RELAY8 is used to send commands to the relay controller. This one-line subroutine does all the hard work.

The RELAY8 subroutine is passed 3 variables to control the relay driver in the follwing form:

RELAY8 *Device, Relay, Status*

Device is a number from 0 to 15 indicating a specific controller to speak to. The device number is set in hardware by jumpers J1 to J4 on the relay controller.

Relay is a number from 1 to 8 specifying which relay to control.

Status can be a 1 for "On" or a 0 for "Off".

Example RELAY8 Calls:

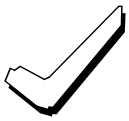
```
RELAY8 0, 5, 1 : Rem Turn On Relay 1 on Device 0.
RELAY8 3, 4, 1 : Rem Turn On Relay 4 on Device 3.
RELAY8 6, 5, 0 : Rem Turn Off Relay 5 on Device 6.
RELAY8 9, 8, 0 : Rem Turn Off Relay 8 on Device 9.
```

The rest of the program is used to ask the relay driver for the status of the 8 relays. It then steps you through decoding the returned value to show you which relays are on.

Sending the command to turn on or off relay 8 three times consecutively triggers the relay and reports the status of all 8 relays.

You will be able to strip many extras out of this program, but at least you get a working example of how to control the relay driver, and how to decode the returned relay status.

One last thing, the command *Print #1, CHR$(X);* sends data out the serial port of your computer. Your computer can send a maximum of 256 ASCII characters, so X has a value of 0 to 255. When an NCD device is set to Device 0, it only listens to ASCII characters 0 to 15. The ";" at the end of this command is required in QBasic, but not in Visual Basic.



Combining several NCD devices on the serial port of your computer may require the RSB serial booster. See the current NCD catalog for details.

Combining with Other NCD Devices:

When controlling other devices, it is important to use the correct driver routine for the device you plan to call. In the following example, we will show how to control the R85/R810 relay board connected as address 1 (jumper J1 on the relay board should be installed). The M1 motor controller in this example is connected to the same serial line coming from the computer. The M1 is set to address 0 (all jumpers removed). Additionally, we will assume the NCD-LCD module is connected to the same serial line. It is set for address 2 (jumper J2 should be installed).

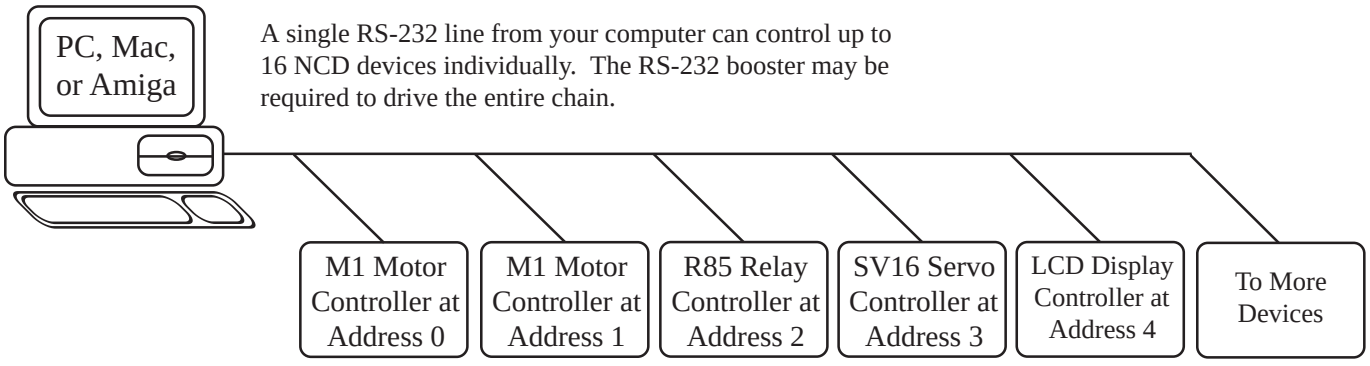
```
M1 0,1,63           : Rem Tells controller named 0 to turn the motor forward (1) at speed 63.
RELAY8 1,5,1        : Rem Tells relay controller named 1 to turn on the 5th relay.
RELAY8 1,5,0        : Rem Tells relay controller named 1 to turn off the 5th relay.
LCD 2,1,"Hello"     : Rem Tells LCD display named 2 to display on line 1 the word "Hello."
END                 : Rem Ends the program.
```

Avoid communication errors to NCD devices by using a quality cable. The NCD serial booster provides the drive current required to control 16 devices over a distance of several hundred feet. Also, make sure the proper routines are being used for each device. Addressing is very important--make sure devices are not set to conflicting addresses and that addresses are set correctly in software.

Applications:

Now that you are familiar with NCD and the R8 series relay controller, we would like to invite you to combine our devices on a single serial line. Up to 16 NCD products may be combined on a single serial line of your computer. Combine the R85/R810 with the LCD controller to display relay status information. Combine with a stepper controller for the precise motor movement found in robotics. Bring an entire robot to life by adding an SV16 servo controller to the network. You can mix and match whatever type of controller you need without limiting yourself to the restrictions of one controller.

Try combining the AKT ASCII keypad transmitter and the IRTR infrared transceiver to send commands to your computer, then have your computer send commands to any device you want, controlling any of 16 devices. All of our manuals are put together in the same easy-to-read format--just put the routines for each device into a single program. Call the appropriate routine using the appropriate device name and you will have reliable control of up to 16 different NCD devices, all from the serial port of a single computer. Watch for the arrival of many more products in the years to come.



We welcome suggestions and recommendations. Please tell us about any errors or omissions you may find in this manual. All recommendations become the property of NCD Inc.
The R85 and R810 Relay Controllers are property of National Control Devices.

National Control Devices
P.O. Box 455
Osceola, MO 64776

Phone: (417) 646-5644
FAX: (417) 646-8302

www.controlanything.com
www.controleverything.com
E-mail: ncdryan@aol.com