SIK-232

USER MANUAL

Corporate Headquarters 11711 W.Sample Rd.
Coral Springs, FL 33065

Phone: (800) 770.4700 / 954.545.9000
Fax: (800) 548.5885 / 954.545.9011

www.vutec.com

1. Introduction

The world of home automation and integration has been growing at an enormous rate in recent years. Control has been integrated using everything from a universal remote control to a complete home automation system to a personal computer.

The R2D7 is an interface that bridges from the RS-232 protocol to a single or multiple RP-Busses. RS-232 is a common protocol that is “spoken” by every personal computer and most home automation systems. There are many big names in the home automation industry: Crestron, AMX, Control4, Savant, Vantage, LiteTouch, & Lutron are considered among the largest. In some cases, an additional component will be needed by the home automation system in order to use RS-232.

Using an R2D7 allows up 420 hardwired RP60’s or RQ60’s and 5,940 RP60’s or RQ60’s using a combination of hardwired & wireless controls.

2. Installation

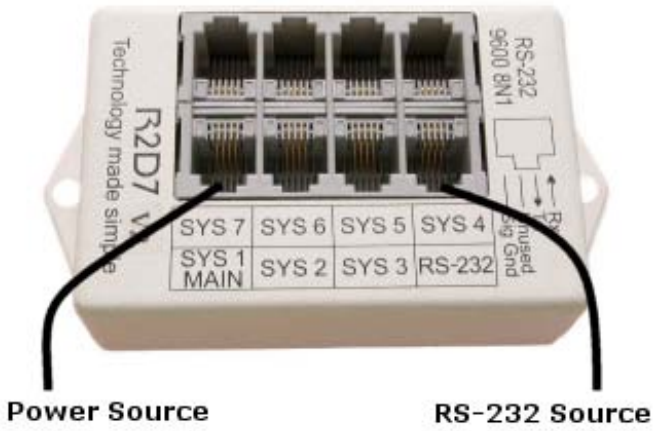


Figure 2.0.1: Basic R2D7 Setup

2.1. Providing a Power Source

A power source must be provided to the R2D7 on system 1. This source may be either a DCPM or an "EYE" port of and RP or RQ controller (see figure 2.0.1).

2.2. Providing an RS-232 Source

The RS-232 source may be any device that is capable of communicating using the standard RS-232 protocol. As mentioned before, this may include devices such as a home automation system or a personal computer.

When configuring the RS-232 device, the following table defines the port settings required by the R2D7:

Port Speed	9600 baud
Data Bits	8
Parity	None
Stop Bits	1
Flow Control	XOn / XOff

To make the cable from the RS-232 source, use the figure 2.2.1. Note that the "Receive" and "Transmit" pins on this diagram are the "Receive" and "Transmit" as labeled on the RS-232 source, not the R2D7.

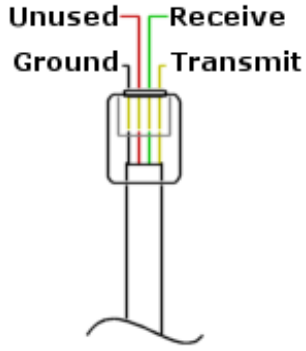


Figure 2.2.1: RS-232 Cable Configuration

2.3. Adding Hardwired Systems

When going from system ports other than system 1 on the R2D7, the connection may go to either an “EYE” or an “AUX” port on the RP60.

2.4. Adding Wireless Systems

The R2D7 has the ability to control motors wirelessly using Radio Frequency (RF) modules when in radio mode.

Plug the RFTM module into system 7 on the R2D7 using standard RP-Bus cable (see figure 2.4.1).

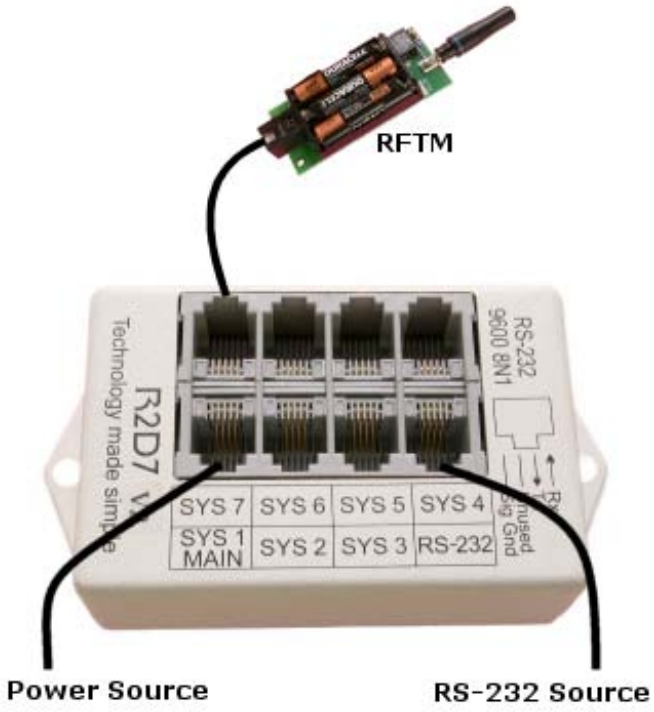


Figure 2.4.1: R2D7 Radio Mode Setup

For more detail on configuring the R2D7 for wireless systems, see section 4.1 “Radio Mode.”

3. R2D7 Protocol

Communication with the R2D7 is defined in two parts: command strings and response strings.

3.1. Command Strings

Strings are not cases sensitive to the R2D7, but for our purposes, we will use lower case for operating strings and upper case for configuration strings.

3.1.1. Operating Strings

Operating strings for the R2D7 are segmented into 5 or 6 parts:

*	1	o	04	020	;

Each command must begin with a "header" character. This character is always the asterisk: "*"

The next segment is either one or two characters. This represents the system address, or the physical port we are addressing on the R2D7 device. If the R2D7 is in radio mode (default), then this number may be anything from 1 through 99. When in bus mode, however, this number may only be 1 though 7. To send the "ALL" command, use 0 (zero) for the system.

Next, we need to give the R2D7 an instruction as a single character. In our example above, we have issued the 'Open' command. The following table defines the commands available:

o	Open (sends the open command to the system for the specified unit address)
c	Close (sends the close command to the system for the specified unit address)
s	Stop (sends the stop command to the system – unit address is ignored)
w	Wind (sends the wind command to the system – unit address is ignored)
p	Program a Motor Control (puts the RP60 in programming mode)
a	Program an Accessory (puts the accessory in programming mode)
q	Quit Sending Current Command (only applicable when sending a "forever" command)

Now we need to indicate which unit the command is intended for. This is an address from our pool of 01 through 60, or 00 to represent "ALL." The address here is the same as if the command were issued from a remote control; the R2D7 is simply another command source. The address may control a group of shades, single shade, and even all shades. Perhaps the address for certain controls indicates running to an intermediate stop.

The next three digits are an optional timing instruction for the R2D7. The value represents the duration of the command in 20^{th}'s of a second. For example, 010 would be half a second. If this information is not specified, then the R2D7 will send the command for 2 seconds. If 000 is specified, then the command will be sent forever.

Finally, the terminator must be sent. This can be either the semi-colon (as in our example) or "CR" (the "Enter" key).

3.1.2. Configuration Strings

Configuration strings for the R2D7 are segmented into 3 parts:

*	V	;

As with operating instructions, the first character is the header. This is always the asterisk ("*").

Next is the setting for the R2D7. The following is a table of valid commands:

V	Print Version (followed by mode: R = Radio, B = Bus)
R	Radio Mode (put system 7 in Radio mode)
B	Bus Mode (put system 7 in Bus mode)

Finally, the terminator is either a semi-colon or the "CR" character.

3.2. Response Strings

Although the RP-Bus is not bi-directional, the R2D7 does report status information back to the sender. The following table describes the responses for each event in the R2D7:

Event	Response
Power On	"version, X-on"
Terminator Received	"LF" if good command "U" if command was not understood "CR" after command was relayed to controls
Buffer Overflow	"O"
Buffer In Empty	"XOn" (Ctrl+Q), but only if XOff was sent
Buffer Is Half-Full	"XOff" (Ctrl+s)

4. Operating Mode – System 7

By default, system 7 is configured for radio mode.

4.1. Radio Mode

As previously mentioned, the R2D7 is capable of controlling up to 420 unique motors. When using RF, this becomes 5940 motors.

See “RR24 User’s Guide” and “RFTM User’s Guide” for more information on radio products.

4.1.1. Training the RR433

As with all other radio transmitters, before the RR433 will respond to commands from the R2D7’s RFTM, each RR433 that will be controlled by the R2D7 must be trained.

The first step is to identify the systems that will be controlled by the R2D7. Each group of RP60’s that is bussed together is considered a system (see figure 4.1.1.1). Note that in our example, the three systems are arbitrary. They must be numbered from 7 or higher, but we could have given each system any number we wanted.

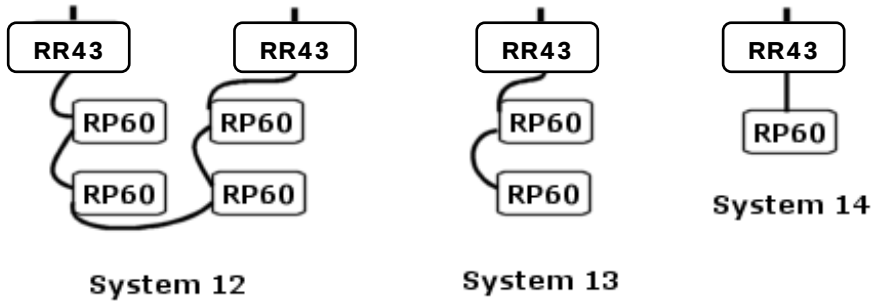


Figure 4.1.1.1: Wireless RS-232 Systems

Once the systems have been identified, commands need to be created from the RS-232 source for each system. In our example, three commands would need to be created, and for simplicity’s sake, we will use the “Stop” command. The three commands would be: “*12s;” and “*13s;” and “*14s;”

For each system, follow the instructions in the “RR433 User’s Guide” to put the RR433’s on that system into “Learn” mode (one system at a time).

Tip:

Especially when dealing with systems containing multiple RR433’s, or when the RR433 is located a long distance from the RS-232 source, it may be easiest to reset the RR433, causing the “Learn” light to remain on until the first signal is seen.

Send the command for that system from the RS-232 source, i.e. send “*12s;” when the RR433’s on system 12 are in learn mode, etc. The lights on the RR433’s in that system should go out. Repeat these steps for each system of RP60’s.

4.1.2. Controlling Motors

Controlling motors in radio mode is identical to bus mode, except that the system number may range from 1 to 99.

In our example from the previous section, we identified systems 12, 13, and 14. So, if we wanted to close all motors on system 13, we would send the command "*13c00;"

Note:
When sending a command to all systems (such as "*0c00;"), the command will **not** be sent to wireless systems. Each system will have to be addressed individually.

4.2. Bus Mode

When in bus mode, system 7 on the R2D7 operates identically to the other ports.

What else do we need to say about bus mode?

5. As a Programming Device

Before attempting to program any devices using the R2D7, read and understand the section “Programming” in the “RP60’s User Guide.”

Devices connected to the R2D7, whether hardwired or wireless, may be programmed using RS-232. This is accomplished by emulating the button sequence performed by hand.

First, the device must be placed in program mode by sending the “p” command for a motor control and the “a” command for an accessory (see section 3: “R2D7 Protocol”) followed by the identification number for the device. Next, the sequence must be sent for the desired feature. Finally, send the “Stop” command to take the device out of programming mode.

Note:

Before doing any programming from the R2D7, be sure to read the manual for the device you wish to program.

Example:

Suppose we wish to set a second group of 6 to motor (main channel) 2 on system 4. We would need to send the following sequence of commands to the R2D7:
*4p02;*4c03;*4o06;*4s;

Example:

Suppose we wish to set a “light” threshold to 75 on an SSPB with the default ID on system 3. We would need to send the following sequence of commands to the R2D7:
*3a01;*3c06;*3o07;*3o05;*3s;

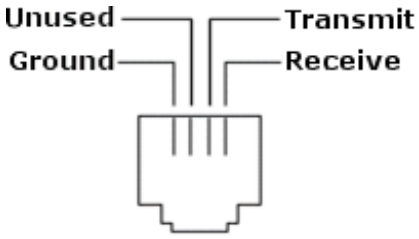
Example:

Suppose we wish to enable the wind lockout for all RP60’s on system 5. We would need to send the following sequence of commands to the R2D7:
*5p00;*5c10;*5o01;*5s;

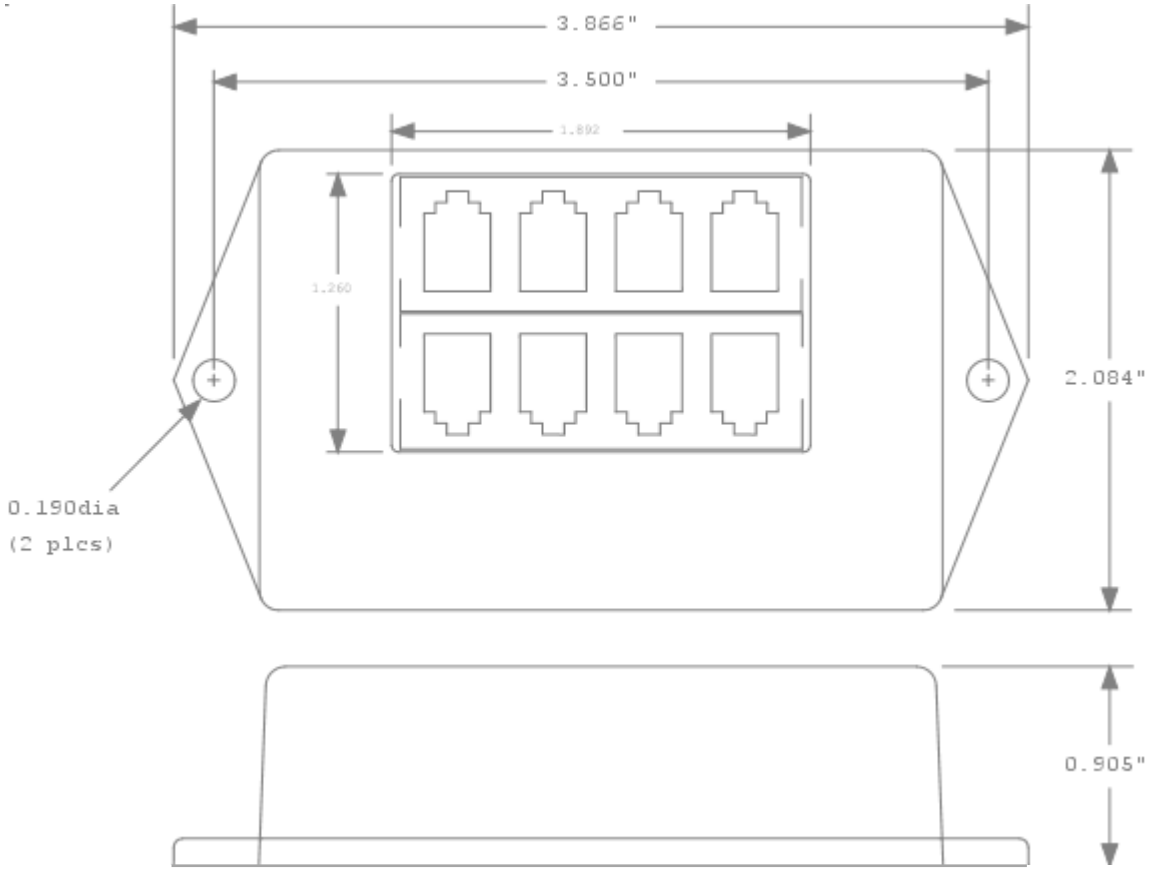
6. Appendix A: Specifications

6.1. RS-232 Port Pin Configuration

When looking at the face of the R2D7, the RS-232 port configuration is:



6.2. Dimensions



7. Appendix B: Quick Reference

7.1. Port Settings

Port Speed	9600 baud
Data Bits	8
Parity	None
Stop Bits	1
Flow Control	XOn / XOff

7.2. Command Strings

Segment	Data	Characters
Header	*	1
System	1 – 7 for Bus Mode, 1 – 99 for Radio Mode, 0 = "ALL"	1 or 2
Command	o, c, s, w, p, a, q	1
Address	01 – 60, 00 = "ALL"	2
Timing (optional)	001 – 999, represents 20^{th}'s of a second, 000 = forever	3
Terminator	semi-colon ";" or "CR"	1

7.3. Configuration Strings

Segment	Data	Characters
Header	*	1
Command	V, B, R	1
Terminator	semi-colon ";" or "CR"	1

7.4. Response Strings

Event	Response
Power On	"version, XOn"
Terminator Received	"LF" if good command "U" if command was not understood "CR" after command was relayed to controls
Buffer Overflow	"O"
Buffer In Empty	"XOn" (Ctrl+Q), but only if XOff was sent
Buffer Is Half-Full	"XOff" (Ctrl+s)

8. Appendix C: Examples

8.1. Standard Examples

Open Motor 3 on System 7	*7o03;
Close Group 27 on System 5	*5c27;
Open All Motors on All Systems	*0o00;
Stop All	*0s;
Open All Motors with Main Channel 3	*0o03;
Close All Motors on System 4	*4c00;
Open Motors 1-4 on System 2	*2o01;*2o02;*2o03;*2o04;

8.2. Timing Examples

Open Motors 1 – 4 with a 1-second Delay	*2o01020;*2o02020;*2o03020;*2o04;
Open All Motors, Stop Main Channel 1 After 4 Seconds, Continue Main Channel 1 After Another 2 Seconds	*0o00080;*0o01;*0o01;