

Prolift Advanced Control Box RS232 Guide

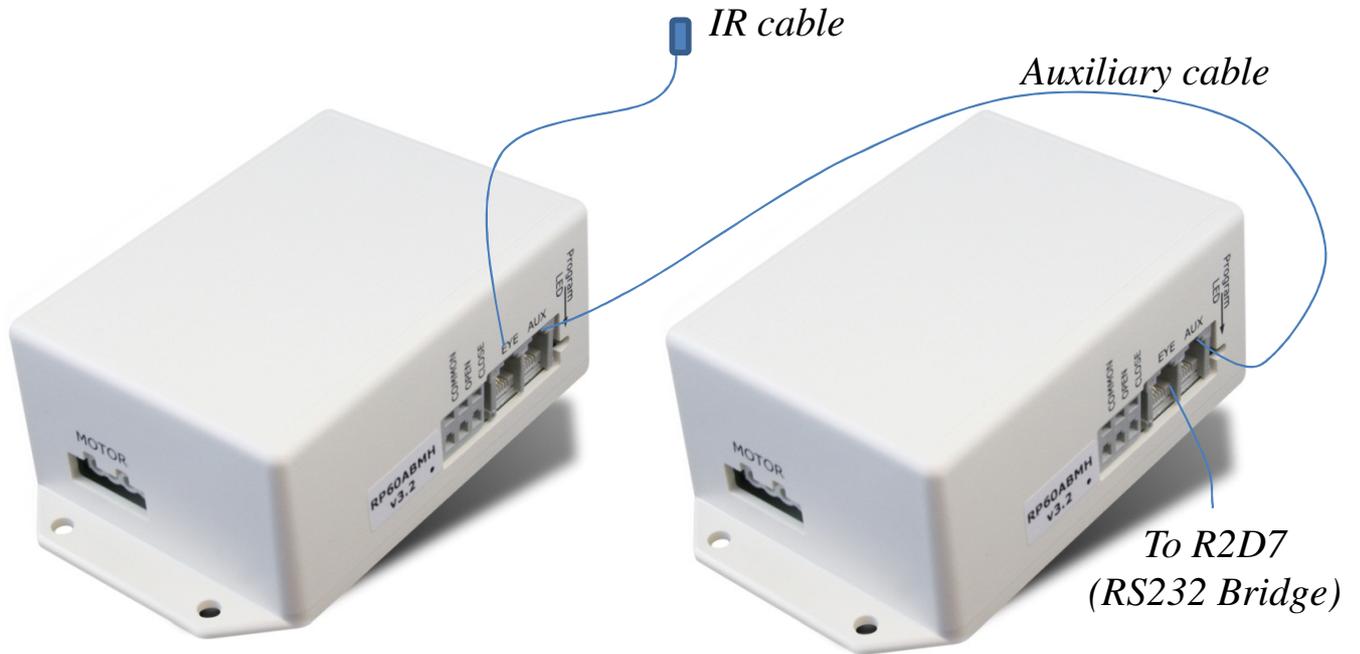


Contents

1. Remote Control Functions and RS232 Command Strings
2. R2D7 V4 Technical Notes
3. R2D7 V4.1 Protocol
4. R2D7 Users Manual



Ultralift Special Instructions (1/2):



Two RP60ABMH are connected in parallel via Auxiliary cable . Input power cables are ganged to ensure the two controllers work at the same time. Please make sure connections are checked before activating the motors. The IR cable can be connected to either controller EYE port. Use the other EYE port to connect to the R2D7 RS232 communication bridge.

Channel 1: view Position 1/ home →

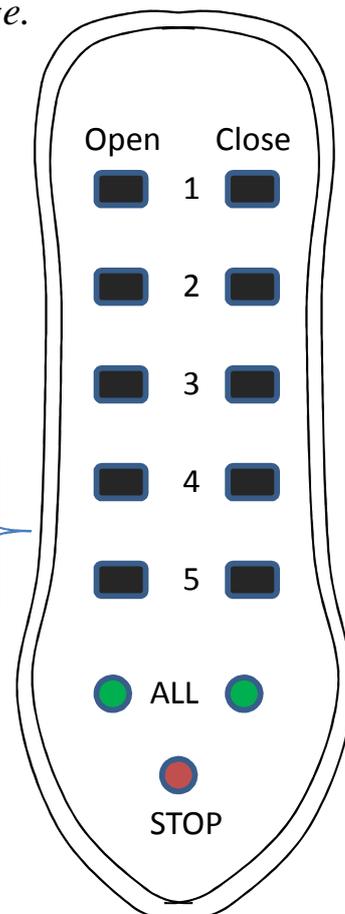
Channel 2: View Position 2/ home →

Channel 3: Service Position/ home →

Channel 4-5: Disabled →

ALL channel: Service / home →

ALL Motor stop →



Channels 1 and 2 are memory positions and when depressed will keep going up to the respective memory positions. To stop the motor, press STOP button or the Channel 3 close button. To jog the motors to either direction, press the Channel 3 open/close button once and depress again to stop.

Ultralift Special Instructions (2/2):

Channel 1: view Position 1/ home →

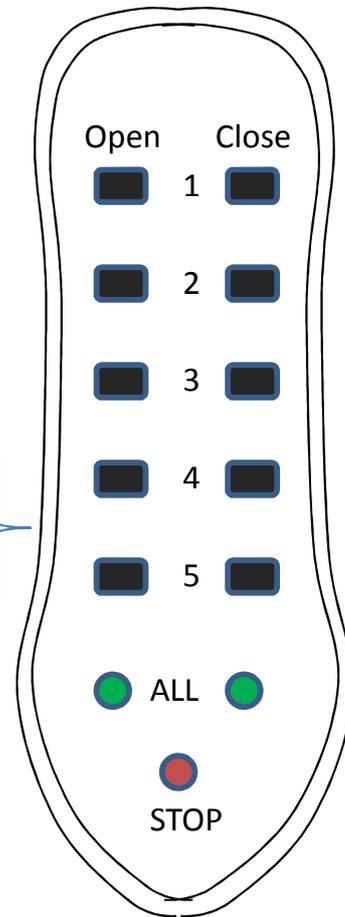
Channel 2: View Position 2/ home →

Channel 3: Service Position/ home →

Channel 4-5: Disabled →

ALL channel: Service / home →

ALL Motor stop →

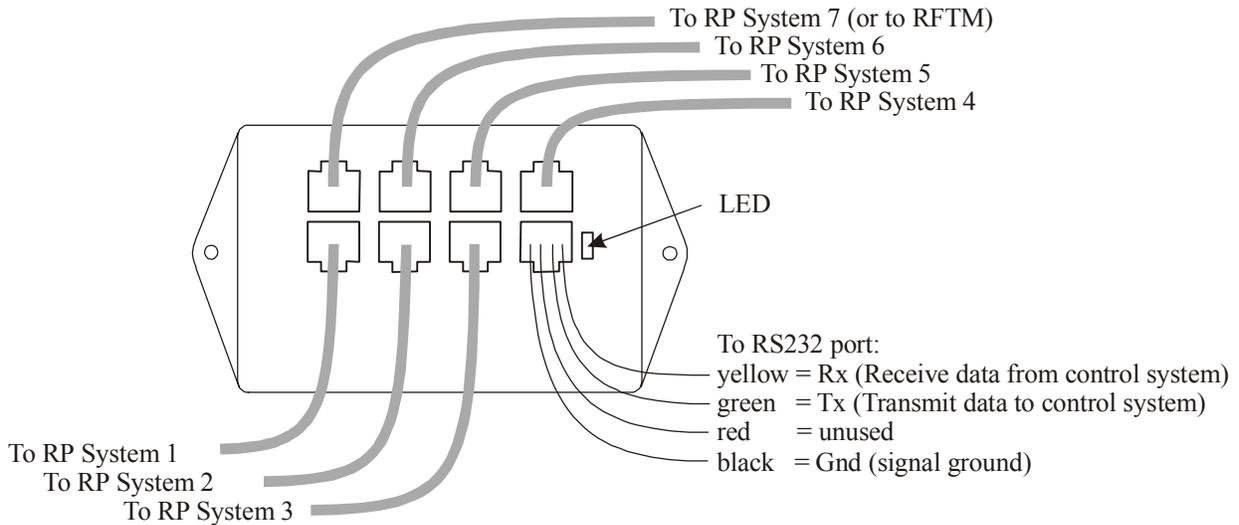


The above remote control button assignments are translated to RS232 command string as shown on the table below. Projector lifts 1 to 4 could be assigned as RP System1 to RP System 4 and fed to a single RS232 source. Refer to R2D7 Tech Notes, Protocol, and User's Manual for further details.

Remote Control Button assignment	Motor action	RS232 COMMAND STRING			
		RP System 1 (Proj Lift 1)	RP System 2 (Proj Lift 2)	RP System 3 (Proj Lift 3)	RP System 4 (Proj Lift 4)
Channel 1 OPEN	View Position 1 Open	*1o01;	*2o01;	*3o01;	*4o01;
Channel 1 CLOSE	view Position 1 Close	*1c01;	*2c01;	*3c01;	*4c01;
channel 2 OPEN	View Position 1 Open	*1o02;	*2o02;	*3o02;	*4o02;
channel 2 CLOSE	view Position 1 Close	*1c02;	*2c02;	*3c02;	*4c02;
Channel 3 OPEN	Service Position 1 Open	*1o03;	*2o03;	*3o03;	*4o03;
Channel 3 CLOSE	Service Position 1 Close	*1c03;	*2c03;	*3c03;	*4c03;
STOP (All Channel)	All Motor Stop	*1s;	*2s;	*3s;	*4s;

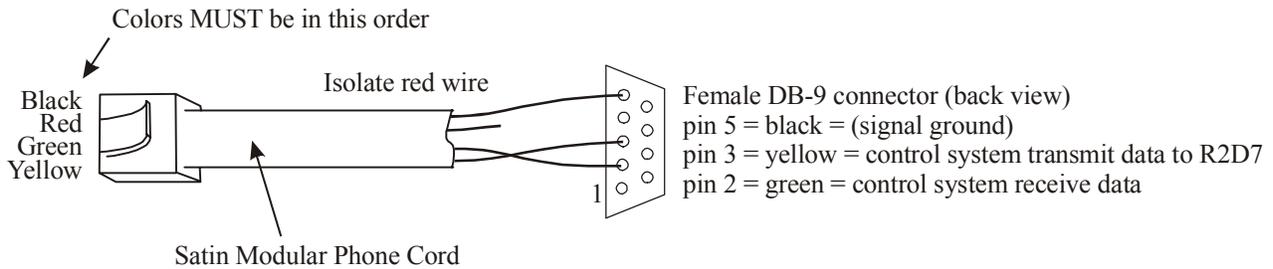
R2D7 V4 Tech Notes

RS232 Port setup: 9600 8N1.



Note that System 1 **must** be connected to the eye port on a functional RP system for this unit to operate, all other system ports may be left unconnected.

A typical cable for connecting to a Home Automation System is as shown:



New to V4:

The LED reports the following:

On power up, it flashes red 4 times if the unit is in Bus mode, it flashes green then red 3 times if Radio mode.

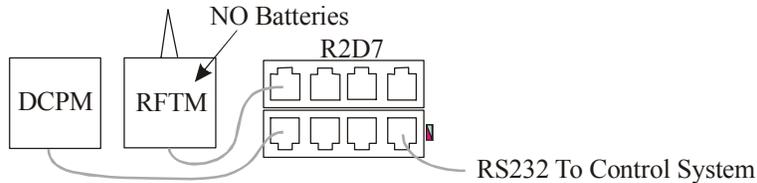
When sending commands on the bus, it flickers green.

If a bad command is received, it will briefly flash red.

The protocol is described in "R2D7 V4 Protocol" document. Case sensitivity has been removed and extraneous characters before a command are ignored. The buffer is much larger.

R2D7 and radio control:

Port 7 has power connected from port 1. This allows an RFTM to be plugged directly in without needing a splitter for the DCPM. Connection as shown is useful when only radio is used.



Examples and other notes:

If you want to emulate the momentary action currently available using a transmitter with an RP in momentary mode, then put the desired RP's in momentary action, and use the following sequence (assuming system 1, channel 23, tilting open):

When the button is pressed, send *1o23000; the R2D7 will begin sending "OPEN23" commands
When the button is released, send *1q; the R2D7 will stop sending when ; is received

The RP will respond appropriately by tilting if short button, latching if held longer than 1.5 seconds.

This method is recommended if an IR transmitter is used in conjunction with the control system.

(If the RP is placed in "Stop on Transmitter Button Release" mode, then the motor will run as long as the button is held)

Another method to accomplish this action is put the RP in maintained action:

When the button is pressed, send *1o23002; the R2D7 will send "OPEN23" for .1 seconds
When the button is released, send *1s; the R2D7 will send "STOP ALL"

The RP will respond by running as long as the button is held.

If the "time to send" value is not sent, then the R2D7 will send the command for 2 seconds. This guarantees the action happens even if the motor is already moving or if it is "momentary action." Stop ALL command is sent for .25 seconds.

A large "scene" can be created by using the ALL systems command. I.e., all motors in group 14 on all systems can be simultaneously opened by sending *0o14;

Complex scenes can be achieved by putting the RP's in "Stop on Transmitter Button Release" mode. The motors will run as long as commands are sent. Use the R2D7 to send timed commands to put the motors at desired locations. Make sure that all motors are fully closed (or open) before doing timed runs!

The RS232 port on the R2D7 interface can be wired with only 2 wires if you choose to ignore the feedback information. This means that the control system programmer must make sure that the buffer never overflows, and that all messages are correctly formatted.

A scene can be sequenced using the R2D7 by sending timed commands with pauses created by sending timed command to an unused channel. This effect can also be done by timing commands inside the control system program.

The R2D7 internal buffer is 256 bytes long. If the buffer reaches half full, then X-off is sent, and the control system is expected to stop sending commands. In this state, when the last message is read out of the buffer, X-on is sent and the control system may resume sending commands. If the buffer overflows, then an upper case "O" is sent back and ALL commands are thrown away and the buffer is cleared. If the control system keeps sending commands during this time, the first one will probably be bad, and the next valid command will be acted upon.

Any command received by the R2D7 that does not conform exactly to the protocol, or if any number is out of range, then an upper case "U" is sent back and that command is thrown away. The exception to this is that extra characters at the front of a command are ignored.

Between every command there is a forced delay of ¾ second. This allows the RP's time to get ready for the next command. (After a stop, the forced delay is only .1 sec.)

When a valid command is received, the R2D7 sends back "LF" to acknowledge the command. It then begins sending the desired command over the IR link. When the timed command is completed, the R2D7 sends "CR" to indicate that it has completed that command and is ready for the next one.

Special characters:

“;” signifies end of command and can be used in place of “CR”. “;” corresponds to ASCII character number 59 (0x3B)

“CR” or <CR> signifies end of command and can be used in place of “;”.

“CR” corresponds to ASCII character number 13 (0x0D).

“LF” signifies “line feed” and is ignored by the R2D7. “LF” corresponds to ASCII character number 10 (0x0A).

“*” signifies start of command and corresponds to shift 8 on most keyboards, ASCII character number 42 (0x2A).

“X-off” (Ctrl-S on most keyboards) requests sender to stop sending, ASCII character number 19 (0x13).

“X-on” (Ctrl-Q on most keyboards) requests sender to resume sending, ASCII character number 17 (0x11).

R2D7 V4.1 Protocol

<i>Serial Port to BUS (RP Receivers)</i>		
<i>Information</i>	<i>Data</i>	<i># of characters</i> ¹
Header (Start of command)	* (asterisk) for use on BUS to control RP	1
Subsystem number or Administration command	0 ² – 7 if BUS, 7 – 99 if Radio V ³ = report version R = make port 7 Radio ⁴ B = make port 7 BUS	1 or 2
Command ⁵	o = open c = close s = stop ¹⁵ w = wind p = program RP a = program accessory q ⁶ = quit sending	1
Channel ⁷	00 ⁸ – 60 ⁹	2
Time to send ^{10 11}	0 0 0 ¹² – 9 9 9 20 th of seconds ¹³ (this value is optional)	3
Terminator	; (semicolon) or “CR” (enter key)	1
Pause	Send a command on an unused channel for the desired pause time.	

<i>BUS to Serial Port</i>	
<i>Event</i>	<i>R2D7 sends</i>
Powered up	“version ³ , X-on”
Terminator is received	“LF” if good command “U” sent if command cannot be understood “CR” is sent After command has been relayed to motors
Buffer ¹⁴ overflows and all data is purged	“O” is sent
Buffer ¹⁴ is half full	“X-off” (Ctrl-S)
Buffer ¹⁴ is empty	“X-on” (Ctrl-Q), but only if X-off was previously sent

Please refer to the “R2D7 Installation Guide” document for additional information including command examples.

¹ Total message length may be as short as 2 and as long as 10 characters, typically 6

² 0 means ALL systems (for super groups) ALL does not include the radio channels

³ Version reports “4.1” then r if administered as Radio, or b if BUS, followed by carriage return (please note that all characters after V in this command are ignored)

⁴ Radio requires a RFTM be plugged into port 7 jack without batteries

⁵ Case is not important: s is the same as S and so on

⁶ Stop sending a forever command. Any other command will stop transmission, but an error will be reported

⁷ If command is stop or quit, then this value is optional and ignored

⁸ 00 = ALL

⁹ 60 sometimes called “cut strap ALL”

¹⁰ If no digits, then the command is sent for 2 seconds, except for stop, program, or wind which are sent for ¼ second

¹¹ Time between commands = .75 sec if multiple commands are buffered (Time = .1 sec after a stop)

¹² 000 = Forever, use the q command to stop transmission

¹³ Max timed pulse = 50 sec. If this is a radio command, the value is rounded up to the nearest 10th second

¹⁴ Buffer is 256 bytes long

¹⁵ Stop is a “universal stop” command. There is no channel associated with this command – see examples below.

Repeat of a command (open or close) can generally be utilized to achieve a ‘stop’, depending on the features programmed into the RP60xyz motor control.

R2D7 V4.1 Protocol

<i>Special characters</i>		
<i>Character</i>	<i>Description</i>	<i>ASCII Character Number</i>
“ ; ”	Semicolon, signifies end of command	59 (0x3B)
“ CR ”	Carriage return, signifies end of command	13 (0x0D)
“ * ”	Asterisk, Shift 8 on most keyboards	42 (0x2A)
“ LF ”	Line feed, ignored (not used)	10 (0x0A)
“ X-off ”	Ctrl-S on most keyboards, requests sender to stop sending	19 (0x13)
“ X-on ”	Ctrl-Q on most keyboards, requests sender to resume sending	17 (0x11)

Example strings:

System 1 Close ALL for default 2 seconds: ***1c00;**

System 2 Open ALL for default 2 seconds: ***2o00;**

System 3 Close 5 for .75 seconds (momentary shade possibly): ***3c05015;**

System 4 Open 9 for 5 seconds: ***4o09100;**

System 5 Close 23 for default 2 seconds: ***5c23;**

System 6 Open 49 for .25 seconds (momentary shade possibly): ***6o49005;**

System 7 Close 60 for default 2 seconds: ***7c60;**

All Systems Open 21 for default of 2 seconds: ***0o21;**

System 1 Stop for default .25 seconds: ***1s;**

R2D7

User's Manual

Version 1.2 – 2004 May 20

**Corporate
Headquarters**

11811 Upham Street, #2
Broomfield, Colorado 80020
USA

Phone: +1 303.469.9322
Fax: +1 303.469.5129

1. Introduction.....	3
2. Installation	4
2.1. Providing a Power Source	4
2.2. Providing an RS-232 Source.....	4
2.3. Adding Hardwired Systems	5
2.4. Adding Wireless Systems	5
3. R2D7 Protocol.....	6
3.1. Command Strings	6
3.1.1. Operating Strings	6
3.1.2. Configuration Strings	7
3.2. Response Strings.....	7
4. Operating Mode – System 7.....	8
4.1. Radio Mode.....	8
4.1.1. Training the RR24	8
4.1.2. Controlling Motors	8
4.2. Bus Mode	9
5. As a Programming Device.....	10
6. Appendix A: Specifications.....	11
6.1. RS-232 Port Pin Configuration	11
6.2. Dimensions	11
7. Appendix B: Quick Reference.....	12
7.1. Port Settings.....	12
7.2. Command Strings	12
7.3. Configuration Strings	12
7.4. Response Strings.....	12
8. Appendix C: Examples.....	13
8.1. Standard Examples.....	13
8.2. Timing Examples	13

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, 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 and 5940 RP60's using a combination of hardwired & wireless controls.

2. Installation

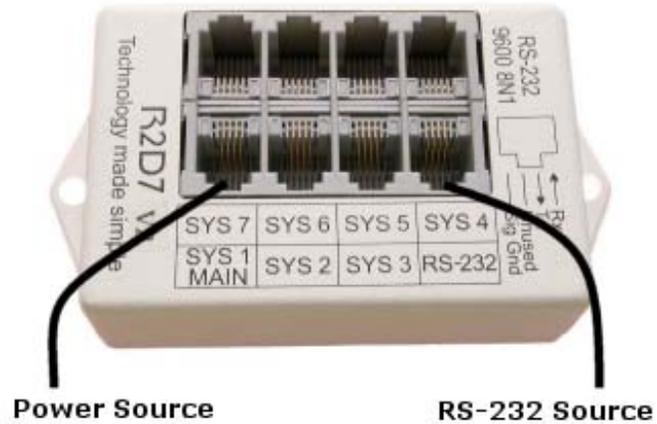


Figure 2.0.1: Basic R2D7 Setup

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 (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.

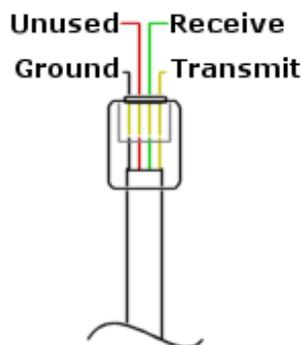


Figure 2.2.1: RS-232 Cable Configuration

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).

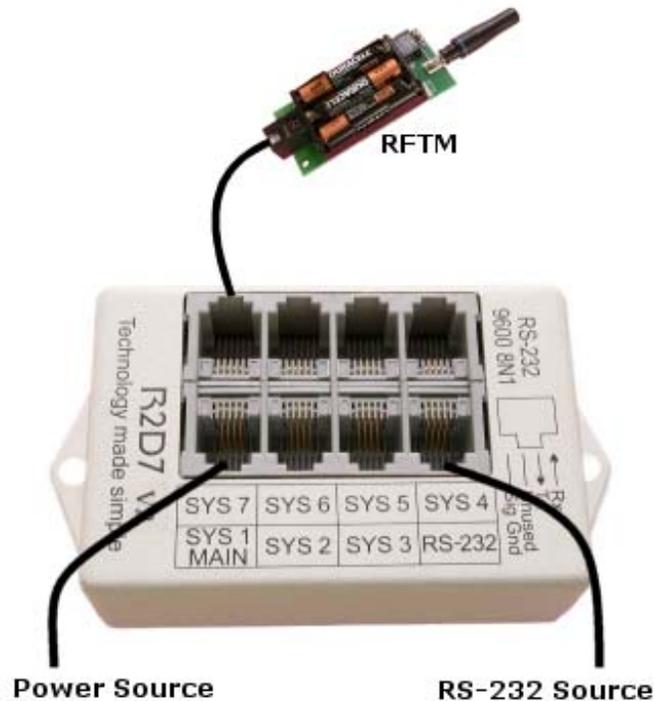


Figure 2.4.1: R2D7 Radio Mode Setup

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 through 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 20ths 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 RR24

As with all other radio transmitters, before the RR24 will respond to commands from the R2D7’s RFTM, each RR24 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.

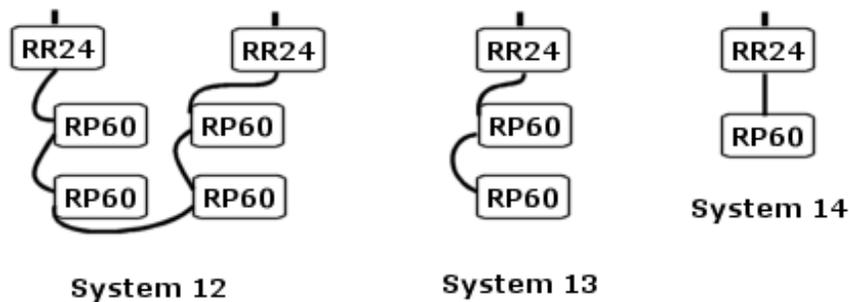


Figure 4.1.1.1: Wireless RS-232 Systems

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 “RR24 User’s Guide” to put the RR24’s on that system into “Learn” mode (one system at a time).

Tip:

Especially when dealing with systems containing multiple RR24’s, or when the RR24 is located a long distance from the RS-232 source, it may be easiest to reset the RR24, 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 RR24’s on system 12 are in learn mode, etc. The lights on the RR24’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:

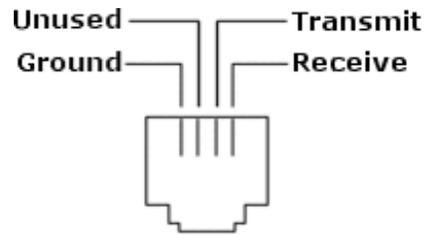
Suppose we wish to enable the wind lockout for all RP60's on system 5. We would need to send the following sequence of commands to the R2D7:

```
*5p00;*5c10;*5o01;*5s;
```

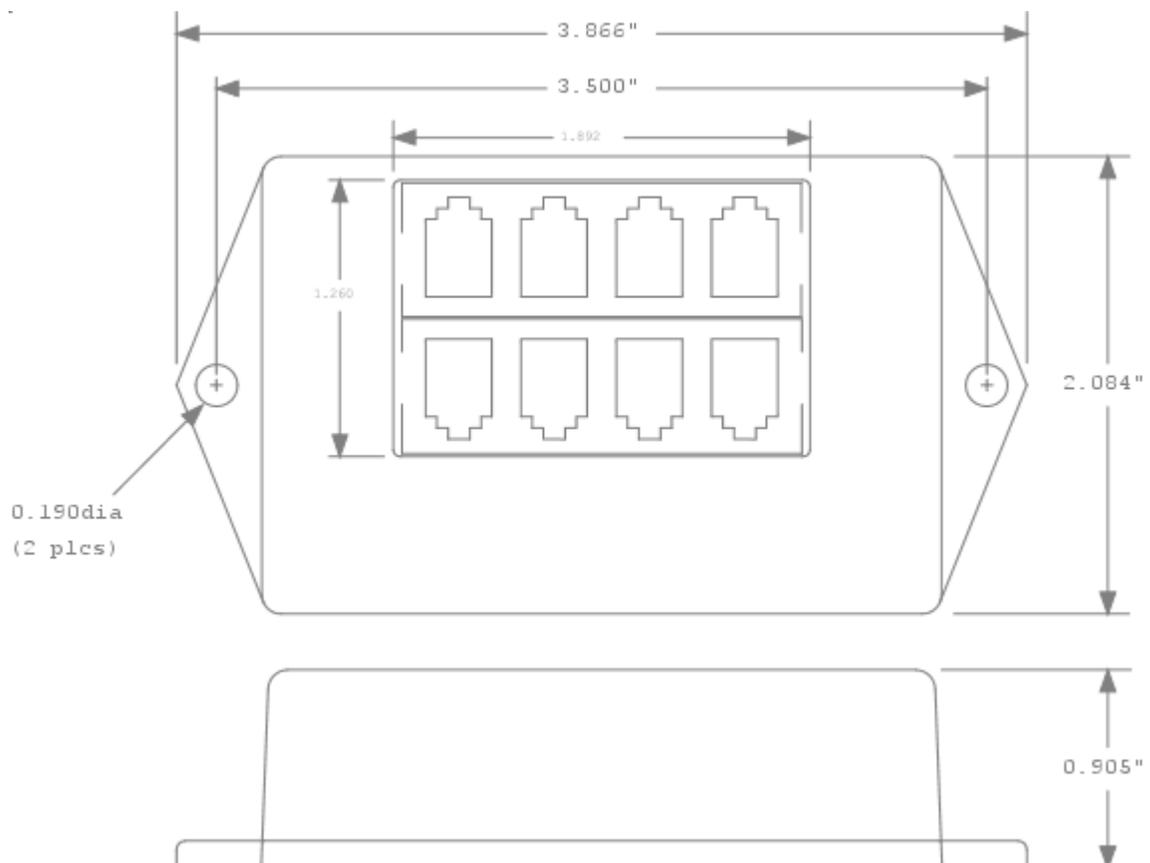
6. Appendix A: Specifications

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;

THANKYOU FOR CHOOSING
**ULTRALIFT
AUSTRALIA**



Ultralift Australia Pty Ltd

10 Vernon Ave, Heidelberg Heights VIC 3081 Australia

Email: sales@ultraliftaustralia.com.au

Web: www.ultraliftaustralia.com.au

Phone: 61 3 9459 0873

Fax: 61 3 9457 6686

Member

