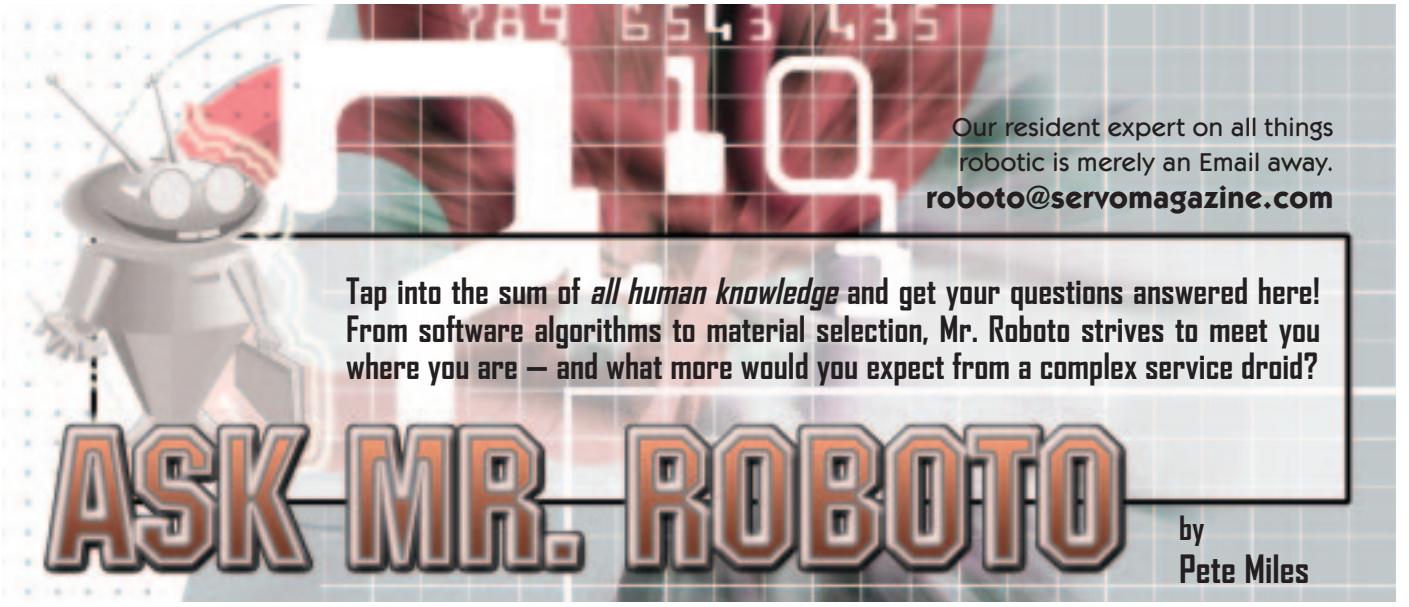


Our resident expert on all things robotic is merely an Email away.
roboto@servomagazine.com

Tap into the sum of *all human knowledge* and get your questions answered here! From software algorithms to material selection, Mr. Roboto strives to meet you where you are — and what more would you expect from a complex service droid?

ASK MR. ROBOTO

by
Pete Miles

Q. I want to build a two-legged walking robot. Right now, I am planning on using the 32 servo controller from Lynxmotion to control all the servos in my robot. It is controlled by a PC using an RS-232 serial cable, but the cable length is going to limit the range of my robot. Is there some way to control the servo controller via some sort of a wireless method instead of using the RS-232 cable? If so, can you tell me how to do it and what I need to purchase?

— Bill
Schenectady, NY

A. With all of the Bluetooth devices on the market today, this turns out to be a relatively simple and inexpensive thing to do. What you are going to need to obtain are two things: A Bluetooth dongle that connects to your computer and a Bluetooth serial modem that connects to the SSC-32 (serial servo controller from Lynxmotion; **www.lynxmotion.com**; see Figure 1).

To demonstrate how to do this, I have chosen the Bluetooth Modem — BlueSMiRF from Spark Fun Electronics (**www.sparkfun.com**) — to be the wireless device that will connect to the SSC-32. This is a direct wireless serial cable replacement that is ideal for robotic applications, especially smaller due to its size. Table 1 lists some of the specifications for the BlueSMiRF modem from the original equipment manufacturer and Figure 2 shows a photo of the modem. One of the nice features of this modem is that it uses a frequency hopping scheme that enables it to be used in harsh RF environments, which is needed with all the different 2.4 GHz devices operating around us these days.

On the computer side of things, I have chosen the Bluetooth USB Module (also from Spark Fun Electronics). This module — also known as a dongle — is about the same size as a standard USB memory stick, so it will work well with laptop computers. Figure 3 shows a photograph of the USB dongle. Make sure that you look inside the package cover for the half-size CD that contains the installation software.

Prior to connecting the BlueSMiRF modem to the SSC-32 and the USB Bluetooth dongle to your computer, you will need to install the software for the USB Bluetooth module on your computer. After the software is installed, plug in the USB Bluetooth dongle into the computer, and a "Welcome to Bluetooth" window will appear. Make sure that the "Use security level medium" checkbox is UNCHECKED, then press the OK

Size	1.9" L x 0.6" W x 0.15" T
Weight	2 grams
Antenna	Built-in
Output Power	12 dBm
Receive Sensitivity	-83 dBm
Range	100 m (300 ft) **
Frequency Range	2.4-2.542 GHz
Supply Voltage	3-6V
Current Draw:	
Receive, RX	40 ma
Transmit, TX	80 ma
Idle	1.4 ma
Average	25 ma
*** See discussion at the end of the article*	

Table 1. Specifications of the BlueSMiRF Bluetooth serial modem.

Figure 1. The 32 servo serial servo controller from Lynxmotion.

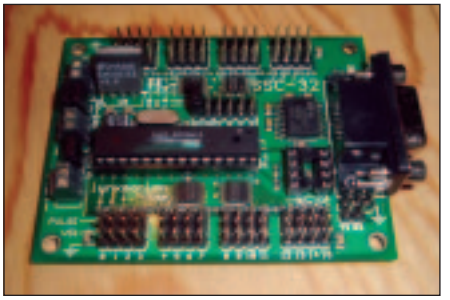
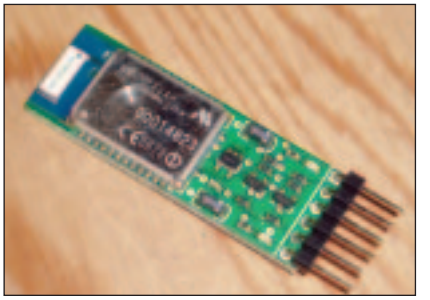


Figure 2. BlueSMIRF Bluetooth serial modem from Spark Fun Electronics.



button. Figure 4 shows the main window that will appear next.

Every Bluetooth device that is in range of this computer and that is recognized as a valid Bluetooth service, will be seen orbiting the "sun." Clicking on the sun will cause it to search for new Bluetooth devices. At this point, power up the BlueSMiRF modem (without connecting it to the SSC-32) with a +5V power source and power ground. Once the BlueSMiRF modem is powered up, click on the sun and after a few moments, an icon with the name "Spark Fun - BT" will be in orbit (see Figure 5).

Figure 3. Miniature Bluetooth USB dongle/module.

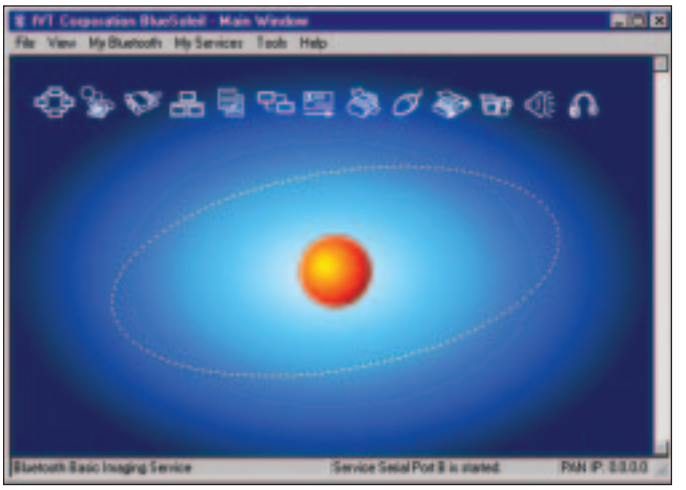


Figure 4. Main connection window for the BlueSoleil software.

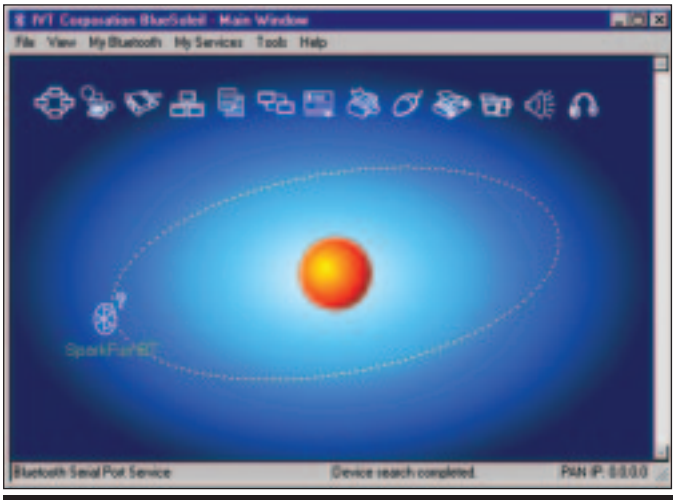


Figure 5. BlueSMiRF modem has been found.

To establish a connection between your computer and the BlueSMiRF, you will need to right-click the Spark Fun - BT icon, select the Connect menu item, and click the Bluetooth Serial Port Service sub-menu item (see Figure 6). Once the connection is made, you will see a dotted line appear between the Spark Fun - BT icon and the sun. The Bluetooth tutorial section on the Spark Fun website provides more information about setting up the Bluetooth software.

There are two LEDs — green and red — on the BlueSMiRF modem. When the modem is first powered up, the green LED will blink with a period of about 2 Hz. The blinking green LED means that it has power, but it is not communicating with anything else. When the connection is made, the green LED will turn off, and the red LED will turn on.

Now connecting the BlueSMiRF to the SSC-32 is a straightforward process. The first thing you need to do is remove the two jumpers on the SSC-32 that are connecting the two TX pins and the two RX pins together, which are located next to the DB-9 connector (the RS-232 cable connector; see Figure 7). Next, using a three wire cable, connect the RX-I pin on the BlueSMiRF to the TX pin on the SSC-32 and connect the TX-0 pin on the BlueSMiRF to the RX pin on the SSC-32. The third wire connects the grounds together on both devices. Always remember, connect the TX (transmit) line on one device to the RX (receive) line on the other device. Since the SSC-32 doesn't use flow control in its serial communications, you will need to connect the CTS-I and RTS-0 on the BlueSMiRF together. If you don't do this, the modem won't work. Table 2 lists the wiring connections, and Figure 8 illustrates how to physically wire the two devices together.

Figure 6. Connecting to the BlueSMiRF modem.

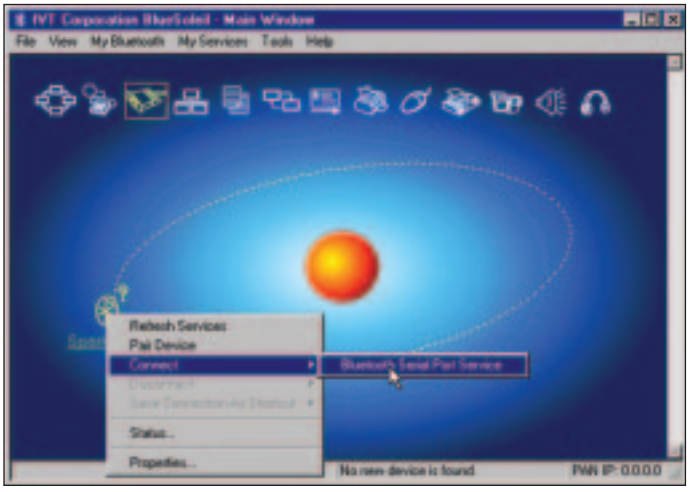
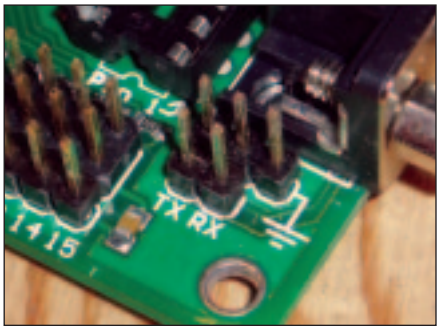


Figure 7. Close-up view of the TX and RX pins on the SSC-32.



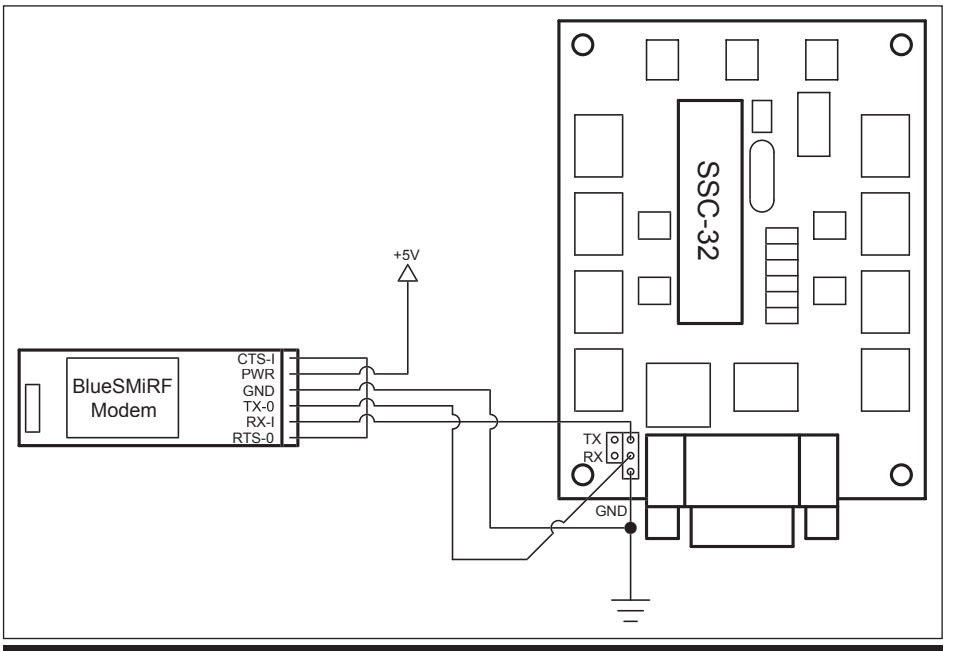


Figure 8. Illustration of wiring the BlueSMiRF to the SSC-32.

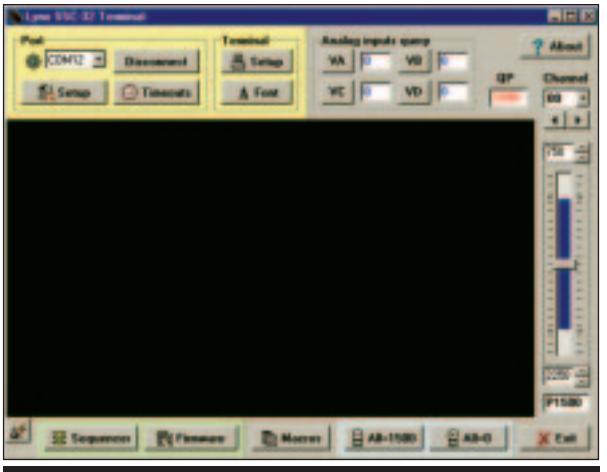


Figure 9. Lynx SSC-32 terminal program, version 1.05.

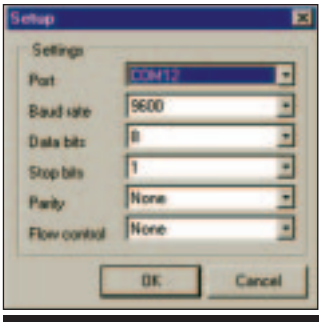


Figure 10. Initial COM port setting for communicating with the BlueSMiRF (port COM12 will be different on your machine).

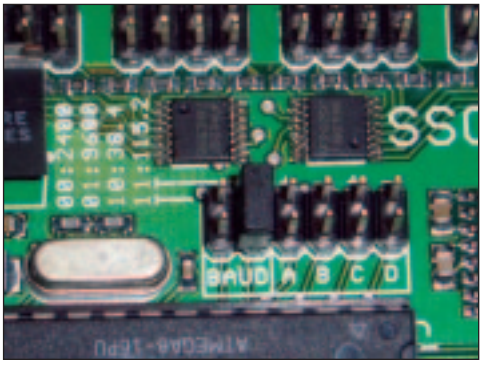
The SSC-32 does not provide any regulated power outputs to power any external devices, so you will need to provide your own regulated 5V power for the BlueSMiRF modem. Do not use the servo power from the SSC-32 to provide power to the BlueSMiRF modem. This may result in unreliable operation due to voltage dips and spikes.

I don't know what software you are planning on using to control the SSC-32, so I will demonstrate how to set up and run the Lynxterm software that is available for free at Lynxmotion. Power up the SSC-32 and the BlueSMiRF modem, establish a communication connection with the USB Bluetooth module, and then start the Lynxterm application. Figure 9 shows the main window for the Lynxterm program. Next, you will need to determine which COM port the USB module is operating on. To do this, right click on the Spark Fun - BT icon on the BlueSoleil program, and then click on the Status menu item. This will show you which COM port you are connected to.

The default baud rate for the BlueSMiRF modem is 9600 bps. So in the Port window in the Lynxterm program, click the Setup button, and change the COM settings to 9600 baud and change the COM port to the actual port your USB module is located. Figure 10 shows what all of the settings needs to be for the initial communication. Note: The COM12 is the COM port my Bluetooth Module was connected to for this demonstration. Your system will most likely be different.

The default baud rate on the SSC-32 is set to 115,200 bps, since the two jumpers are pre-installed on the SSC-32. This needs to be changed to 9600 baud by removing one of the jumpers. See Figure 11 for a close-up view of the proper jumper setting for 9600 baud. The SSC-32 manual explains how to set other baud rate speeds. If the baud rates for the SSC-32, the BlueSMiRF, and the Lynxterm programs are not all operating at the same speed, then the whole system won't work properly.

Figure 11. Jumper settings on the SSC-32 for 9600 baud serial communication rate.



BlueSMiRF	SSC-32	Notes
CTS-I		Connect to RTS-0 on BlueSMiRF
PWR		Connect to +5V Source
GND	GND	Connect to GND of PWR Source
TX-0	RX	
RX-I	TX	
RTS-0		Connect to CTS-I on BlueSMiRF

Table 2. BlueSMiRF to SSC-32 wiring connections.

Since the BlueSMiRF modem doesn't locally echo the characters sent to it, you will have to adjust the Terminal setting of the Lynxterm program to see the characters you type on the screen. Figure 12 shows what the terminal settings need to be to communicate with the BlueSMiRF modem. The

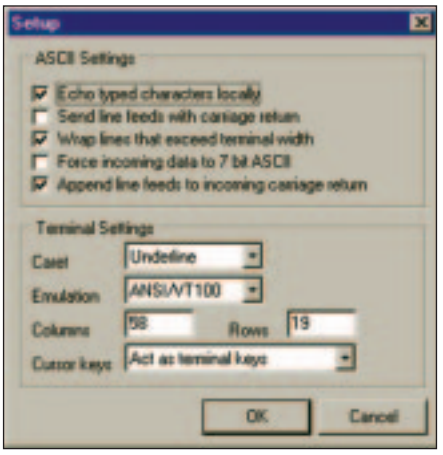


Figure 12. Terminal setup for communicating with the BlueSMiRF modem.

AT Command	Description
+++	Enter Configuration Module
AT	The Attention command prefix, should return OK
ATMD	Exit out of configuration mode
ATVER, ver1	Get BlueSMiRF firmware version
ATSI,1	Status Information: returns Bluetooth address ID
ATSI,2	Status Information: returns modem name
ATSI,8	Status Information: returns current baud rate in HEX
ATFRST	Reset factory defaults
Changing Baud Rate: ATSW20,10,0,0,1 ATSW20,39,0,0,1 ATSW20,157,0,0,1 ATSW20,472,0,0,1	 Change to 2400 Baud (10 ASCII) = (0A Hex) Change to 9600 Baud (39 ASCII) = (27 Hex) Change to 38,400 Baud (157 ASCII) = 9D Hex) Change to 115,200 Baud (472 ASCII) = (1D8 Hex)

Table 3. BlueSMiRF modem configuration and status commands.

Lynxterm program will still be able to control the SSC-32 with different terminal settings, but you won't be able to change the internal configuration settings on the BlueSMiRF modem itself if the settings are different from what is shown in Figure 12. Once all the settings have been adjusted, press the Connect button, and you should be ready to go.

The first thing you should do with the modem is verify that you have good communications between the BlueSMiRF and the SSC-32. On the Lynxterm program, type the word "ver" and press the carriage return key. When you are doing this, you should see the green LED on the SSC-32 servo controller blink after each time a key is pressed on the keyboard. After you press the carriage return button, you should see "SSC32-1.06XE" on the screen (this may change slightly depending on which firmware version you have in your SSC-32). If you don't see anything or some garbage characters, then there is a communications problem. Check your settings or wiring.

After completing all these steps, you are ready to install some servos and servo power to the SSC-32. If all is working right, you will be able to command the servos to move with the Lynxterm software.

If you are controlling a lot of servos at one time, you will probably want to increase the baud rate on the BlueSMiRF. To do this, you will need to make a configuration change in the BlueSMiRF modem.

Now, to change any of the configurations in the BlueSMiRF modem, type "+++" followed by a carriage return to enter the configuration mode. When you do this, you should see OK on the Lynxterm window. While you are in the configuration mode, you will not be able to control the SSC-32. To exit out of the configuration mode, type "ATMD" followed by a carriage return; you will get an "OK" on the screen, and the Lynxterm program will have control of the SSC-32 again. Table 3 lists a small set of the many configuration commands the BlueSMiRF modem has. A complete user guide for changing configurations can be obtained from Spark Fun Electronics.

It is recommended that before making any configuration changes to the BlueSMiRF modem, test that you can enter and exit the configuration mode by typing "+++" then "ATMD" (with a carriage return after each command). You should see an OK after each command. If you don't get this, then there is a communication problem. Remember, the default baud rate for the BlueSMiRF modem is 9600 baud. If you ever forget what you changed the baud rate to, type the ATSI,8 command to see the current setting.

Now for a short discussion on effective communication range. The BlueSMiRF modem is a Class 1 Bluetooth device, which means it should have a maximum range of 100 meters (330 feet) or so. But there are many factors that will affect the actual range that you will obtain. Factors include other 2.4 GHz RF devices operating in the same area, microwave ovens (they operate at 2.45 GHz), obstructions between the transmitter and receiver like walls and doors, and the size of the room. In addition to these, the type of antennas both devices are using, the orientation of the antennas relative to one another, and output power and receiver sensitivity differences between the two devices can factor in.

In my testing, I was only able to get about seven meters of direct line-of-site communication between the USB dongle and the BlueSMiRF. This was surprising since both devices are Class 1 devices. After reviewing the specifications for the USB module further, I found that its maximum output power is only 4 dBm, which (according to Table 4) is the maximum output power for a Class 2 device. It would then have a maximum range of 10 meters. I don't know why the USB dongle packaging advertises that it was a Class 1 device and then lists a contrary maximum output power of 4 dBm.

To make things at bit more

Device Class	Maximum Output Power	Maximum Range
Class 1	100 mW (20 dBm)	100 m
Class 2	2.5 mW (4 dBm)	10 m
Class 3	1 mW (0 dBm)	1 m

Table 4. Bluetooth output power classes.

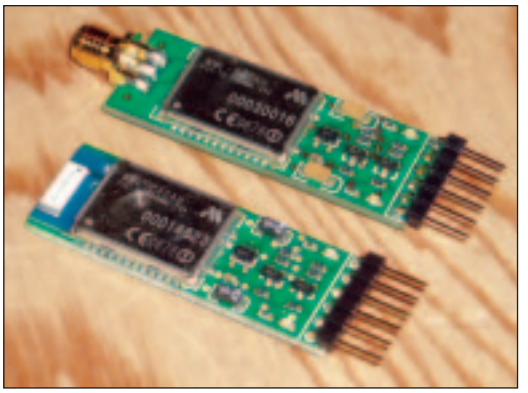


Figure 13. BlueSMiRF modems with built-in ceramic antenna (left) and SMA external antenna mount (right).

interesting, I changed the regular BlueSMiRF modem with a BlueSMiRF RP-SMA modem with a 2.4 GHz duck antenna. Both of these modems are identical, with the exception that the first one has a built-in ceramic chip antenna and the other has an SMA external antenna mount (see Figure 13). With everything else being the same, the transmission range increased to well over 15 meters (the maximum range wasn't tested since I didn't go outside the house to increase the testing range). Though the USB dongle's output power wasn't increased, adding the antenna to the BlueSMiRF improved its reception sensitivity, thus effectively increasing its range.

When long range transmission requirements are needed, the components with the lowest output power rating need to be able to transmit over that range. Otherwise, there may be reliability issues in the data link. In this example, both devices were advertised to be Class 1 devices and advertised to have 100 meter ranges, but in actual testing, the advertised output power rating provided a better indicator of the communication range between the two devices.

A lot of information was presented here, which might give the impression that putting together a wireless robot control system is complicated, but in reality, with the right components, it is almost a plug-and-play. The BlueSMiRF modems are simple and easy replacements for serial cables to put together a wireless communication link. You will probably spend more time reading this article than setting this system up. Here are some interesting notes about the SSC-32 and the BlueSMiRF modem that can help diagnose setup problems:

• The active baud rate on the SSC-32 is determined by the jumper settings at power-up. Changing the jumpers after power-up doesn't change the baud rate settings on the SSC-32.

• When a physical RS-232 serial cable is connected to the SSC-32, the servos will not move if the baud rate settings between the Lynxterm program and the SSC-32 don't match. (Obvious, but see what follows.)

• When the baud rate settings of the BlueSMiRF modem and the SSC-32 are matched at power-up, the baud rate setting on the Lynxterm program can be anything you want it to be, and you can still control the SSC-32.

• When the baud rate settings of the BlueSMiRF modem and the SSC-32 do not match at power-up, the servos will not move regardless of what the baud rate settings are on the Lynxterm program. But, the internal configuration of the BlueSMiRF modem can still be changed by the Lynxterm program. Note also, the green LED on the SSC-32 will still blink as it receives data. It just won't process it.

• After the SSC-32 is powered up, changing the BlueSMiRF modem's baud rate to match the baud rate of the SSC-32 will enable the Lynxterm program to control the servos.

• After power-up, changing the BlueSMiRF modem's baud rate to something that is different from what the SSC-32 is set at disables all servo motion control. The LED on the SSC-32 will not blink when receiving commands.

• Baud rate changes on the SSC-32 require a power-reset to take effect. **SV**