ARRL Periodicals Archive – Search Results
A membership benefit of ARRL and the ARRL Technical Information Service

ARRL Members: You may print a copy for personal use. Any other use of the information requires permission (see Copyright/Reprint Notice below).

Need a higher quality reprint or scan? Some of the scans contained within the periodical archive were produced with older imaging technology. If you require a higher quality reprint or scan, please contact the ARRL Technical Information Service for assistance. Photocopies are $3 for ARRL members, $5 for nonmembers. For members, TIS can send the photocopies immediately and include an invoice. Nonmembers must prepay. Details are available at www.arrl.org/tis or email photocopy@arrl.org.

QST on CD-ROM: Annual CD-ROMs are available for recent publication years. For details and ordering information, visit www.arrl.org/qst.

Non-Members: Get access to the ARRL Periodicals Archive when you join ARRL today at www.arrl.org/join. For a complete list of membership benefits, visit www.arrl.org/benefits.

Copyright/Reprint Notice

QST Issue: Feb 2007
Title: A Simple BluetoothT Radio Interface
Author: Lynn Hansen, KU7Q

Click Here to Report a Problem with this File

A Simple Bluetooth™ Radio Interface

Bluetooth wireless devices are springing up everywhere, especially among cellular telephone users. Now even your transceiver can "go Bluetooth"!

Lynn Hansen, KU7Q

For years I've been intrigued by the idea of using a small personal data assistant (PDA) to control my limited selection of ham radios. In 2001, I decided to create a small program to do just that. The result was named *CTR817* (short for Control The Radio FT-817). This program ran on a Pocket PC (PPC) and was written in Microsoft's *eMbedded Visual Basic*. It was limited to controlling the Yaesu FT-817 transceiver and required a serial connection between the PPC and the radio. The resulting cabling and adapters took up more space than the PPC!

Where's My Serial Port?

Since 2001 many changes have occurred in the PPC market. Faster processors and more memory were added. Serial ports were deemed "legacy ports" and many manufacturers discontinued providing them on their devices. To fill the connectivity gap, USB, WiFi and Bluetooth options have been added. Development software has also become more powerful. Microsoft supports the PPC with its *Visual Studio .NET Compact Framework* programming environment.

Late in 2004, Klaus Ludwig, DL7AIR, contacted me with an idea that would replace the serial port needed for *CTR817*. He had developed a small infrared transceiver that would plug into the ACC port of the FT-817. All I needed to do was to update the program to support IR and the world would fall back into equilibrium.

As with almost everything connected to computers, this task ended up being more complicated than originally thought. It was easy enough to enable the IR port on the PPC, but to my surprise I learned that not only were there several flavors of IR drivers, every PPC manufacturer used a different registry key to identify their particular IR port. Since we used the "raw IR" driver, errors introduced because of a noisy connection translated into bad commands to the '817. If you've ever played with the control port on an '817, you know the radio does not like bad commands. It usually locks up and requires a hard reset to awaken it. It didn't take long before the IR transceiver found its place on the "well, it looked like a good idea" shelf.

Bluetooth to the Rescue!

All was not lost, however. As mentioned previously, PPC manufacturers started adding Bluetooth technology to their devices. In order to support Bluetooth, I decided to completely rewrite *CTR817* using *Microsoft Virtual Studio .NET*. With a high-powered development environment, the gloves came off. Support for ICOM, Kenwood, Yaesu and Elecraft radios was added, Bluetooth, TCP/IP and WiFi, call sign lookup, DX cluster monitoring, and K1EL Winkey[1] support. CTR-Remote[2] was born.

Bluetooth technology uses a limited range (30 to 100 meters) spread spectrum radio for connectivity and provides a wide range of services. It can wirelessly transfer files between two computers and it supports two-way audio. A Bluetooth enabled cell phone allows you to wirelessly use your wireless phone. Now that's confusing!

Bluetooth technology also supports virtual serial ports. Now we're getting somewhere. A Bluetooth serial connection needs a host and a client device. The PPC's Bluetooth driver provides the host; all we need to do is give it something to talk to.

[1]Notes appear on page 69.
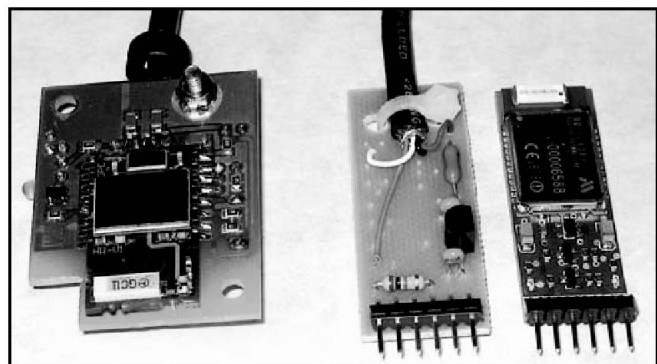

Figure 1 — The original CTR-BlueAir.

Figure 2 — The original CTR-BlueAir PC board and the new CTR-BlueLync interface with BlueSMiRF module.

Klaus and I developed a Bluetooth based radio interfaced dubbed CTR-BlueAir (Figure 1). This small device used a pre-built Bluetooth module from Amber Electronics[3] in Germany. The original version used surface-mount devices and was sold as ready to use. Adapters were developed to interface the Yaesu port to ICOM and Kenwood interfaces. A special version was developed for the ICOM PCR-1000 receiver.

Building and programming each unit proved to be very time consuming so we redesigned the device using through-hole components and offered it in kit format.

Even though the CTR-BlueAir kit version proved to be popular, it still required an excessive amount of time to assemble parts, machine the box and program the module. In addition, module costs were increasing and shipping costs from Germany were excessive. It was time to look for a new solution.

The Third Time is the Charm

What was needed was a simple interface that didn't require a lot of machining or programming. A solution was found at www.sparkfun.com.[4] Their BlueSMiRF Bluetooth RF module is a perfect fit for this application. It contains essentially the same features as the Amber module but in an easier to use format. It can be user programmed and doesn't require any special skills to install or use. Figure 2 shows an original CTR-BlueAir unit next to the new Yaesu interface board and a BlueSMiRF module.

BlueSMiRF has a built-in antenna and it is FCC Part 15 approved. Being a Class 1 Bluetooth device, it has a range of about 100 meters. It's also available with an external SMA antenna connection for even greater range.

Now that we have a device, all we need is an interface. The RF module provides TTL serial I/O, just what the '817 likes. When it arrived, I powered it up with a 5 V power supply, wired its I/O lines to the '817's ACC port and controlled the radio using *CTR-Remote*! Surely, it can't be that easy... but it was.

That could be the end of this story, if this were a Yaesu magazine, or I were a dyed-in-the-wool Yaesu ham. But this isn't a Yaesu magazine and in addition to the '817 I own Kenwood TS-680 and ICOM IC-7000 transceivers as well as an ICOM PCR-1000 receiver. It was obvious that more needed to be done.
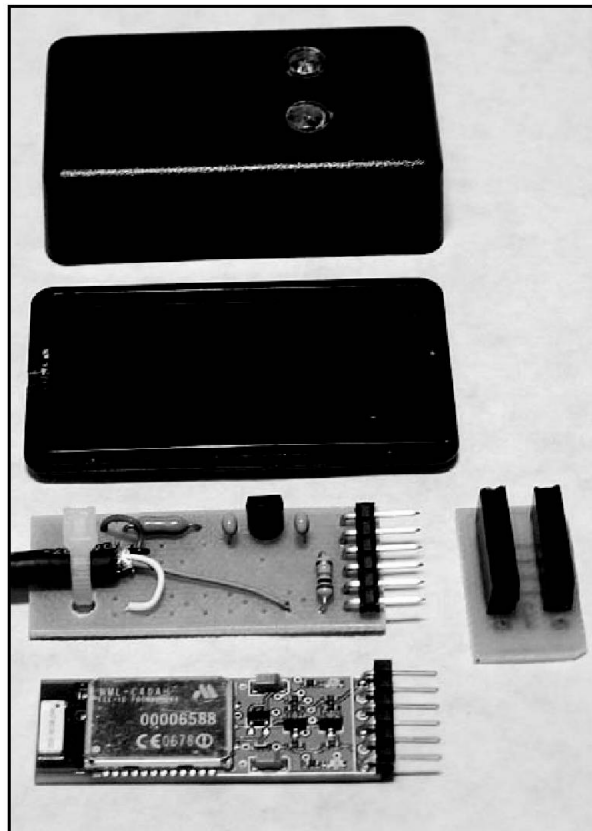

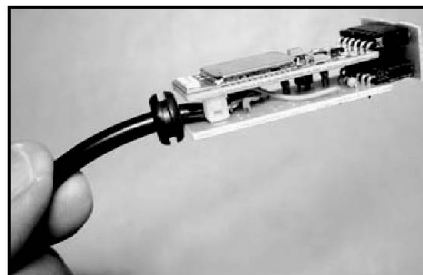Figure 3 — The Yaesu interface ready to assemble.
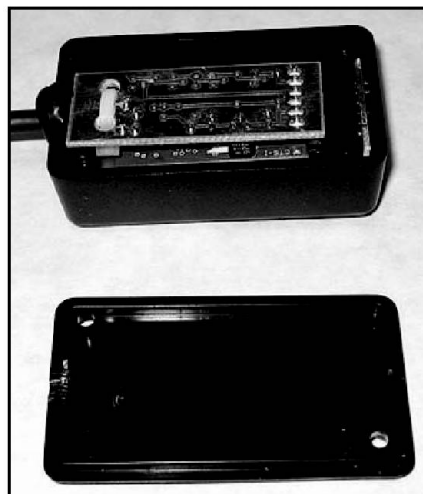

Figure 4 — A side view of the Yaesu interface.

Figure 5 — The interface nestled in its box.

Bluetooth for Everyone

The RF module has a rather odd shape, at least for a radio interface. It also has a 10 V dc maximum input voltage, not ideal for today's dc-powered rigs. These two problems were solved using another circuit board of similar design to house the interface electronics and a small interconnect board to connect the two together. CTR-BlueLync was born!

This simple design provides several benefits. First, you only need one RF module. The radio interfaces are inexpensive, self-contained and easy to build. Essentially, it's a smart cable. You can build the ones you need and share one RF module, saving considerable cost.

Another nice feature is that as radio designs change or new ideas come along, new interfaces can be designed and built.

A single interface board design is used for Yaesu, Kenwood, ICOM (including the PCR-1000) and Elecraft K2 radios, depending on which parts are installed.

A Yaesu interface is shown in Figure 3 ready to be assembled in its Hammond box. Both boards plug into the interconnect board (the small board with two 6-pin vertical connectors) and the cable exits the rear of the box through a grommet. Lenses are installed on the bottom of the box to allow the status LEDs to be seen.

A side view in Figure 4 shows the compact assembly. Although it is small, it is not difficult to build. The package fits nicely into a Hammond 1551H 60 × 35 × 20 mm enclosure as seen in Figure 5.

Let's Build It

Construction is straightforward and virtually any construction technique can be used. I am selling assembled units and kits, but you can also buy your own BlueSMiRF RF module directly from www.sparkfun.com and assemble the necessary interface yourself. Interface schematics and PC board layouts are available at my Web site.[5] A perf-board can also be used. If you choose to go this way, use one with solder pads spaced at 0.1 inch intervals and use 30 gauge wire-wrap wires to connect the circuit.

First, decide on the radio interface. This is usually the easy part, unless you have a lot of radios. If you use the PC board to build the device, place the parts and solder them in. The perf-board takes a little more planning, but you can use the parts placement on the PC board as a guide. Regardless of the construction method used, keep the components
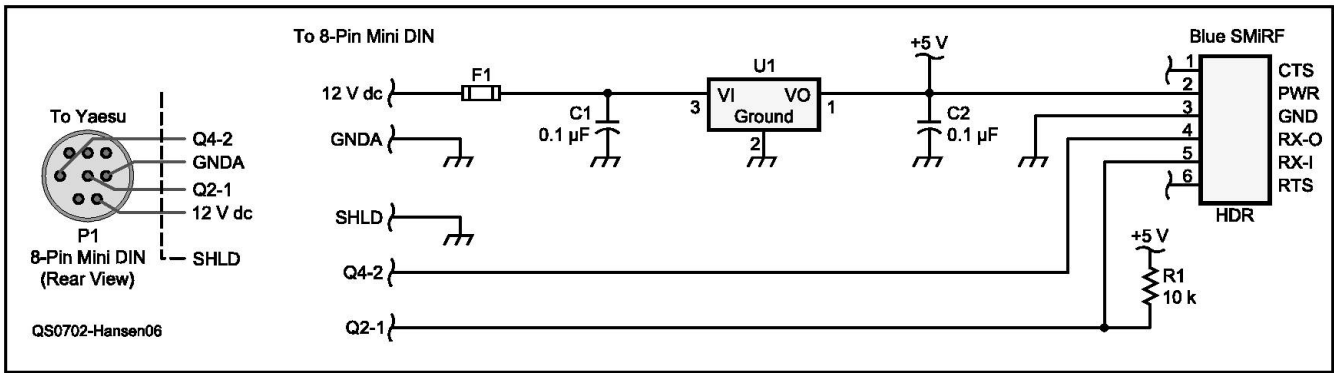
Figure 6 — A schematic diagram of the Yaesu interface.

C1, C2—0.1μF 50-V capacitor.
F1—Pico fuse; 250 mA.

HDR—6-pin 90° pin header.
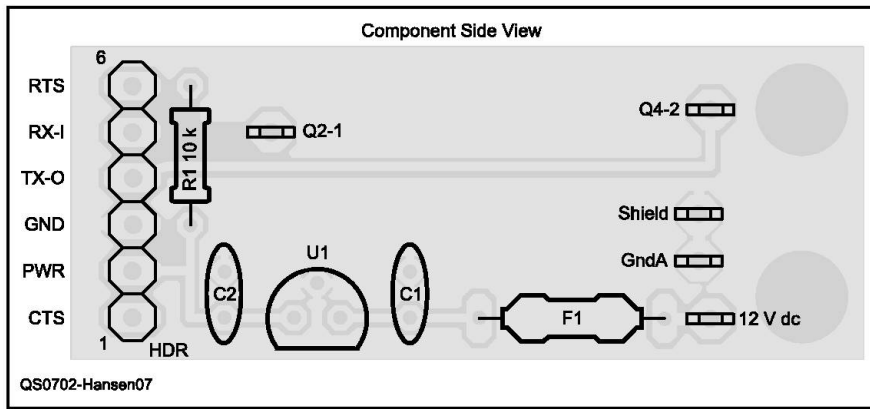R1—10 kΩ, ¼-W resistor.

U1—78L05 5 V dc voltage regulator.

Figure 7 — Yaesu PC board layout.

as close to the board as possible and clip their leads short as there is not a lot of clearance between the interface and RF boards.

If your chosen interface uses a FET, be careful. Observe static precautions when bending the FET leads and installing them on the board. FET devices are sensitive to static and can be easily damaged if not handled appropriately.

The schematic for the Yaesu interface is shown in Figure 6. The board layout is shown in Figure 7.

Two 6-pin vertical low-profile connectors on the small interconnect board provides sockets for the two boards and routes signals between the boards. The interconnect board was originally designed for a 14-pin DIP socket. Mount the 6-pin vertical connectors in the first 6 holes on the board (the holes with traces). Leave the two holes with no traces empty.

Be especially careful when soldering the parts to the printed circuit board. The spacing between the pads and the ground plane is very close and it's easy to bridge solder across the two, shorting out the signal. Once the board has been built, inspect each trace for bridges by holding the board up to a light or using an ohmmeter to measure the resistance between each trace and the ground plane.

The trickiest part of the construction process involves wiring the interface connector. The DIN and miniDIN connectors can be a challenge. Take your time and use heat-shrink tubing between each pin. Surplus Macintosh serial cables can be used for the 8-pin mini-DIN Yaesu interfaces.

Once the interface is complete, plug the interface into the interconnect board and power it up with an external 12 V dc power supply. Check for 5 V on pin 2 of the interconnect board. Do this before you connect the RF module, unless you really want to buy another RF module! Once you're satisfied with the interface, power it down and plug the RF module into the interconnect board.

If everything goes as planned, the green LED on the RF module should start flashing after you re-apply power to the assembly. This indicates that the module is listening for a connection request. Next, connect to it using the host service on your PPC (usually called the *Bluetooth Manager*). Once connected, the green LED stops flashing and the red LED lights solid. The RF module initially comes up named *BlueRadios* and does not require a passkey. *CTR-Remote* and *CTR-Rcvr* software provide options to change the module's name, baud, and power output. You can use a terminal program to change these parameters if
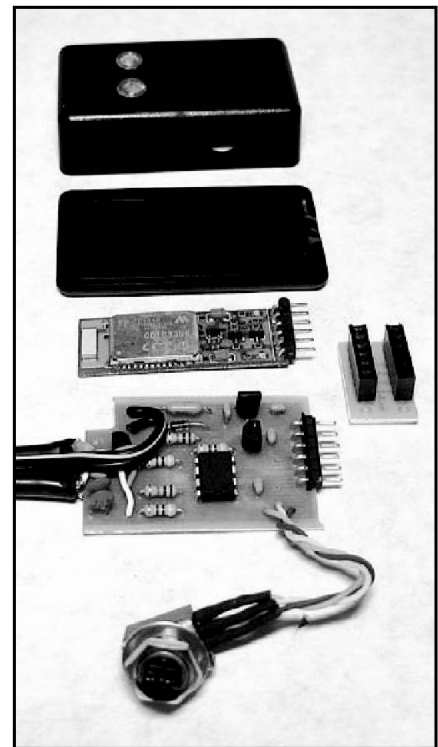

Figure 8 — The Bluetooth keyer interface.

you use other radio control software. Details on how to do this can be found on the Spark Fun Web site.

If you use the Hammond box, trim the inside gussets off the screw posts to allow the boards to sit in the box properly. You'll also need to notch the end of the box for the cables to exit. This notch should be about ⁵⁄₁₆ inch wide and ⅜ inch deep. If you ordered the RF module with the external antenna, extend the cable notch down to the bottom of the box. A nibbler tool makes cutting the notch a simple task.

Drill two ³⁄₁₆ inch holes ⅜ inch from each side and ¹¹⁄₁₆ inch from the end farthest from the cable exit in the box to mount the lenses.

Figure 9 — Using the Bluetooth keyer interface with an Elecraft KX-1.

Figure 10 — Left to right: BlueSMiRF RF model and interconnection board; Yaesu TTL interface; ICOM PCR-1000 RS232 interface; Kenwood TTL interface; ICOM CI-V interface; K1EL Winkey interface.

These lenses allow you to see the status LEDs on the BlueSMiRF. The lenses are made by cutting off the ends of two clear push pins. Use clear epoxy or silicone sealer to glue them in place.

Once everything is complete, plug the unit into your radio's control port and connect to it using your radio control program. Be forewarned, not all control software supports Bluetooth connections, especially older programs. Contact the software developer to verify your program is compatible.

Note: The PCR-1000 version will not work with any software other than *CTR-Rcvr* because other programs automatically ramp-up the serial port rate from 9.6 to 38.4 kbps after they connect. They do this by software command to the radio. Unfortunately, the RF module isn't aware this is going on and its transfer rate stays at 9.6 kbps causing the connection to fail.

Other Options

As mentioned previously, a major advantage of this method of construction is that new interfaces can be easily implemented without the need of purchasing additional RF modules. A keyer module has been developed based on the K1EL WinKey IC. This module is slightly larger than the radio interface modules, but still fits within the Hammond box. The keyer provides a wireless full-featured keyboard/keypad keyer which supports external paddles, side-tone, and a speed adjustment pot. A new tab has been added to *CTR-Remote* to support this keyer. I believe this to be the world's first Bluetooth enabled keyer (Figure 8). There's something almost magical about sending Morse using the handwriting recognition pad on the PPC.

The keyer interface can be used with any solid-state keyed radio. Figure 9 shows it being used with an Elecraft KX-1 transceiver. As you can see, it's easy to put together an extremely compact computer-aided portable station.
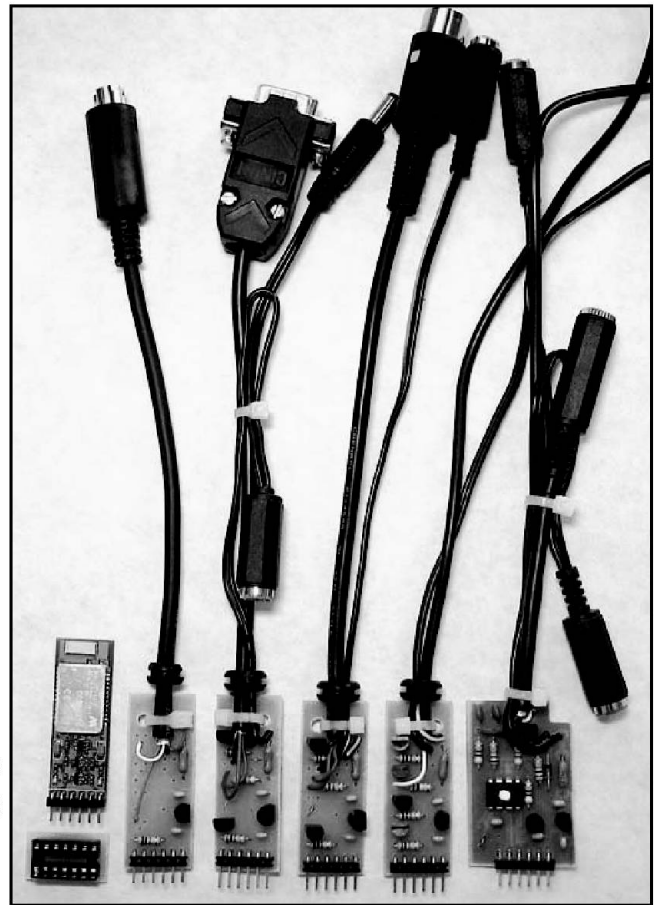
As of this writing, a number of modules are available. Several are shown in Figure 10. As a side note, if you're interested in robotics, the Yaesu interface can be wired such that it can be used to control the Roomba robot vacuum. Details on all of these interfaces can be found on my Web site.[6]

What About Audio?

The BlueSMiRF RF module in its present form doesn't support audio. However, there are several options that can provide remote audio. For receive only, an FM transmitter designed for MP3 players can be used. This works very well when taking your Icom PCR-1000 mobile.

For full transmit and receive the Jabra A210 or similar interface for non-Bluetooth cell phones can be used as the host device for a standard Bluetooth headset. Simply connect the transmit and receive audio on the A210 to your radio's speaker and mic. Several users have used a pair of 450 radios to build a remote audio link. Use VOX to key your radio or software PTT in *CTR-Remote* if your radio supports this function.

Wrapping It Up

I hope you find this project useful. It has finally evolved to the point where it can leave home, so to speak. Build it, experiment with it; come up with new ways of using it. This is why ham radio has always intrigued me. There is a seemingly endless variety of possibilities.

I'd like to thank Klaus for his encouragement and support over the last few years. I'd also like to thank the many hams I have met all over the world that have encouraged and supported this effort. Without their insight and suggestions this project would still be on the "one of these days I'm going to do that" shelf.

You can contact the author at 476 W 3925 N, Pleasant View, UT 84414; **ku7q@arrl.net.**