Lynn Hansen, KU7Q

10914 Westchester Rd., Lava Hot Springs, ID 83246; **lynn@lynovation.com**

Custom Front Panels with a Twist

The author reveals a unique method of designing and building very low cost custom front panels.

Anyone who builds their own equipment can tell you that packaging their project is by far the most challenging part of the process. Adding requirements that the finished product must look good and function well compounds the challenge exponentially. A quick search of the ARRL archives for front panel construction ideas reveals that this subject has generated quite a few articles over the years.

In this article I'm going to review the process that I came up with to create a professional looking custom front panel for my latest project **[1]**, see **Figure 1**. Not only does the front panel look professional, it is extremely inexpensive, costing around $2.00 in quantities of 5. In addition, it includes a built-in cable management system to connect the controls on the front panel to the main circuit board, meaning no more rat's nest of wires between boards!

The Catch

About now, you're probably wondering, "what's the catch?" There's always a catch and this method is no exception. The catch is; you may need to learn something new! You see, the front panel on my SO2R controller is a printed circuit board (PCB) and to create a similar panel for your project may require you to learn how to run a schematic capture and PCB layout program. If you already have a program that does this, and allows you to create custom footprints (the component layout patterns for the PCB), you're ready to start. If not, I highly suggest you look into Kicad **[2]**, a free schematic capture/PCB layout program that runs on just about any operating system out there.

I'll be using Kicad throughout this article to demonstrate my techniques. Installing and running Kicad is outside the scope of this article but there are plenty of online resources available to teach you everything you need to know about the program.

My Front Panel Overview

As mentioned, the front panel on my project is a PCB. The top layer of the PCB is blank — the copper layer is not etched and is at ground potential. It has a black solder mask and white silkscreen lettering. You can choose from several solder mask colors for your project. The silkscreen lettering is pretty rugged and should wear well. You can get the silkscreen in any color you want… as long as it's white — or black on a board with

Figure 1 — Completed CTR2-Mini SO2R Controller featuring the PCB front panel created using the techniques described in this article.

a white solder mask. A close up of the front panel is shown in **Figure 2**.

Make sure the front copper layer on your board is connected to the bottom copper layer with through plated holes or vias **[3]**. If it is not, the board house will build the board without a top copper layer or solder mask, meaning you're front panel is going to be a pretty shade of raw fiberglass with white silkscreen labels.

One cravat with using Kicad is that the font used for the silkscreen text is fixed and can't be changed. You can resize it. If you really want to use another font you can generate your labels in another program then import them as footprint files. For control panel lettering I find the default font perfectly fine.

The controls and LEDs mount to the bottom of the PCB. The bottom copper layer contains pads and traces that connect the leads from each control to other controls or to the 0.1" pin headers that connect this board to the lower board. On my project, inexpensive 8" DuPont female/female jumpers provide the 24-conductor wiring loom between boards.

I designed custom footprints that provide soldering pads for each control and LED. These will be discussed later on. Short jumper wires connect the terminals on each control to the pads on the PCB as shown in **Figure 3**. You can use insulated stranded wire like I did here or you can use bare #26 AWG solid wire from CAT5 cable to make these connections.

One of the requirements of this design is that no through-hole component leads are allowed to spoil the front face of the panel. This necessitated using SMT resistors and capacitors. Two 12-pin SMT 0.1" pin headers were also used in place of normal through-hole pin headers. **Figure 4** shows an SMT header and an LED before soldering. To solder the header to the board just apply a small dab of solder to one pad. Position the header and reheat the solder on that pad. Once you're happy with the position of the header, apply solder to the rest of the pads. To install the LEDs, just bend their leads at a 90° angle and trim off the excess lead from each side. Drop it into the hole on the panel and solder it to the pads. Pay attention to the anode and cathode lead location before you trim them so you don't install the LEDs backwards.

If you're unsure about working with SMT components most board houses will provide SMT components and assembly service for a small charge.

Because this is a strict single-sided PCB, careful layout was required to eliminate all crossover traces. I found it beneficial to layout the signals on the pin headers in the order of the controls they connect to so unobstructed runs could be used. Once this board was laid out, the pin assignments on the header were used to design the lower board. That board is double-sided so it's much easier to route traces over each other.

The most tedious part of assembling this board is cutting, tinning, and installing the 1" jumpers. But even this is way easier than creating a point-to-point wiring loom to individually connect each control to the bottom board.

Let's Get Started

Packaging your project should be part of your initial design effort. While you're brainstorming your latest gee whizzer gizmo think about how you — or another user — will use it. What kind of user interface will it

Figure 2 — Close up of the completed front panel PCB.

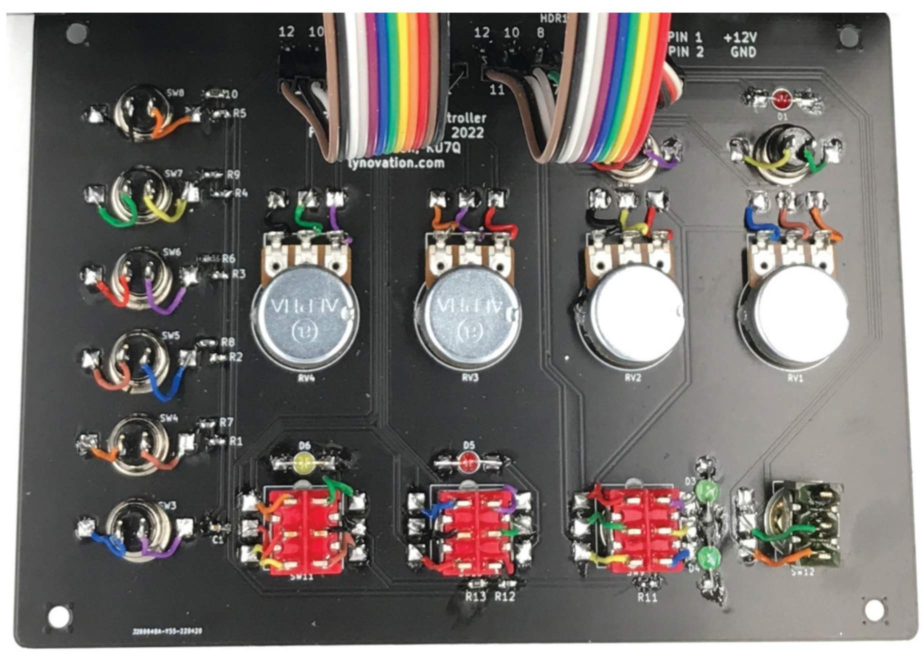

Figure 3 — Bottom of the front panel board showing the design, component wiring, and method of cable management

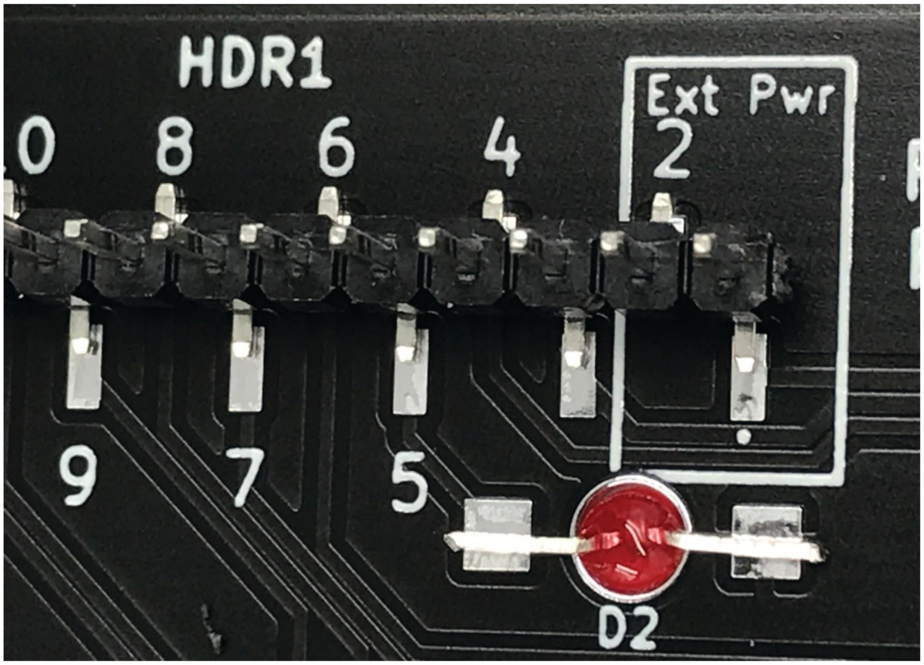

Figure 4 — Close up of an SMT .1" pin header and LED before soldering.

have and where will the controls be located? Will they be mounted on a vertical panel, a horizontal panel, or a sloped panel? Perhaps you're duplicating a project found in *QST*. In that case you already have an idea of how you want to lay out the front panel based on how the original appears in the article. But there's no reason why you can't add your own touches to the design and rearrange the controls to fit your operating style.

With the physical layout in mind, draw up a sketch of where the controls will be mounted on the panel. Make this drawing as close to scale as you can. This not only gives you an idea of how large your panel needs to be but it gives you a chance to organize your controls in a logical order. Group your controls by function. My controller has four control zones, the Tx Enable zone, the gain adjustment zone, the switching zone, and the function button zone.

While you're working on the panel layout, pay special attention to each control's location, spacing and alignment. Try to keep them balanced so the panel doesn't feel heavy on one side and light on the other. For example, the controls on my controller are laid out in a three-row, five column matrix with equal distances between rows and columns originating from the center of the board. The function buttons on the right side don't line up with the other rows but they do line up with the vertical columns. In my panel design everything has balance, a place, and a purpose. The more time you spend on this aspect of your project the happier you will be with the final result.

Next, start looking for a suitable enclosure. For my project I determined a panel about 6" by 4" was needed to mount 16 controls and 6 LEDs. Supply outlets do an outstanding job of linking manufacturer datasheets to the products in their database. Search through their listings until you find an enclosure that fits your needs. For my project I settled on the PacTec KEU-7 sloped panel **[4]**. According to the datasheet **[5]** this enclosure has a blank front panel insert that is 5.876" by 4.125" with four mounting holes 5.3" and 3.735" apart. Perfect, now I have the dimensions for my PCB.

Let the Fun Begin!

The first thing you'll need to do in Kicad is create a schematic for your front panel. If your project is simple you may be able to locate all of its components on the back of the front panel, but normally your front panel will be separate from the main board and require a wiring loom to connect the two. In this case you'll need two schematics: one for the front panel and one for the main board.

To create your schematic, open Kicad and create a new project. Once created, click the Schematic Layout Editor icon to open Eeschema, Kicad's schematic editor. You'll need some symbol libraries to do anything useful so click the Preferences… Manage Symbol Libraries icon to load what you need. Kicad comes with a huge symbol library and you can also download additional symbols from part suppliers. You can use the generic symbols from the Kicad library to create your schematic then assign custom footprints to each symbol for the actual board layout.

Kicad includes a schematic symbol editor. This editor allows you to create your own symbols or rearrange existing symbols to make new devices. For instance, I need a 3PDP switch for this project. There wasn't a symbol for this switch in the Kicad Switch library so I took an existing DPDT switch and added another pole to it. You can find videos online to learn the basics of editing symbols.

Try to keep your schematic organized, clean, and easy to follow. I try to minimize the use of the global signal labels to make the circuit easier to follow. That said there are times where signal labels help declutter the drawing.

The schematic captures the electrical functions of your device. Once you're satisfied with it, it's time to move on assigning footprints.

Footprints

Footprints are files that describe the physical attributes of the schematic symbol they are assigned to. A footprint must be assigned to every symbol on your drawing. The pin number on the footprint correlates to the pin numbers on the schematic symbol. Fortunately, Kicad comes with a huge footprint library too so you don't have to reinvent the wheel every time you add a new device. Part suppliers are also doing an outstanding job providing footprints along with the schematic symbols for the components they sell. However, when designing a front panel PCB you won't be using 'normal' footprints, you'll need to use footprints I have developed, or create your own. They're not hard to create and the Internet is full of helpful videos explaining how to create custom footprints. I'll cover the basics in the next section.

Creating Your Own Footprints

For this project I needed custom footprints for the pushbutton switches, LEDs, toggle switches, and potentiometers. My footprints consist of a mounting hole for the component and pads to wire the components terminals to.

In Kicad there are several layers, or

attributes, for each footprint object. **Figure 5** shows my potentiometer footprint in the editor. Download the datasheets for the components you will be using to get the dimensions you need for your footprints.

In the potentiometer's footprint the yellow circle in the center defines the hole for the shaft that will be drilled in the PCB. The datasheet **[6]** shows the bushing as being 6.35 mm (¼") in diameter. I made the diameter of this footprint object 7.2 mm and the hole in the middle of it 7 mm. This leaves about 0.1 mm of exposed copper around the hole and added an extra 0.65 mm to the hole diameter to allow for the plating that will be applied to the inside of the hole when the PCB is built.

The gray circle is on the Front Fabrication layer. This circle is 20 mm in diameter and represents the size of the knob I planned to use on each pot. This layer does not appear on the PCB itself but it helps me determine the optimum spacing of each pot.

The violet outline is on the Bottom Silkscreen layer. It will be printed on the bottom of the board and gives you a visual indication of the direction the pot is mounted. The VR## text box will be populated when the program annotates the parts on the schematic. The small circle with the + symbol to the left of the hole indicates the location of the anti-rotation key on the pot. This key is a small extrusion of metal on one side of the pot that drops into a hole on the panel to keep the pot from rotating. I added this symbol to the silkscreen so I know where to drill the ⅛" hole partially through the board for the key to drop in to.

NOTE: Anti-rotation keys are also used on the toggle switch washers to keep the switch from rotating. Don't drill the key holes all the way through the PCB or they will be visible from the top. You may need to trim some of the metal off the key since it's designed to go all the way through the panel.

The three green squares at the bottom are assigned numbers that correlate to the pins on the schematic drawing of the device. They are on the bottom copper layer. The ordering of these numbers is critical as they also indicate the pot's turning direction. As indicated on the datasheet the resistance between pins 1 and 2 is 0 Ω when the pot is fully counter-clockwise. These pads will be connected to other devices or the pin header via traces on the bottom layer. I chose to make the pads 2 mm by 2 mm square to allow ample room to solder the jumper wires from the lugs on the pots. I located them sufficiently away from the pot's body to allow room to work with the jumpers.

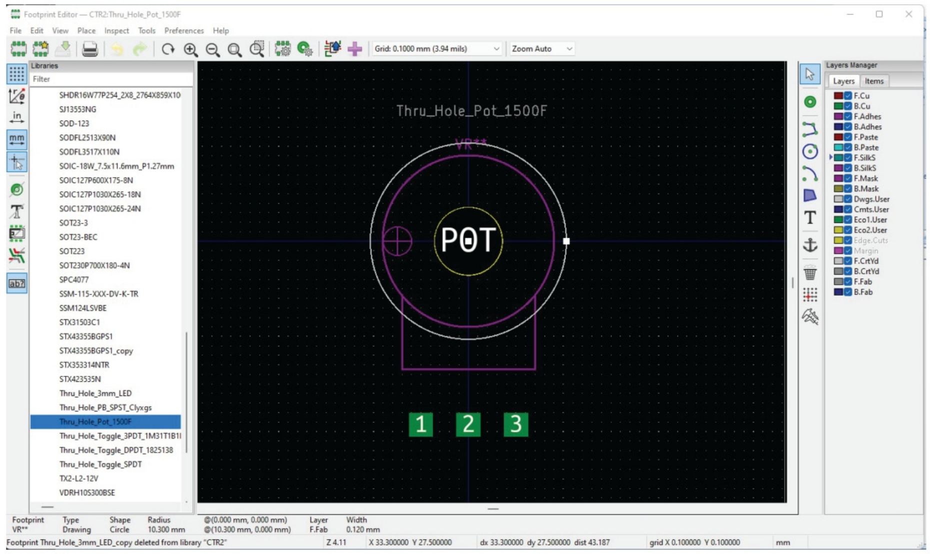

Figure 5 — Screen shot of Kicad Footprint Editor.

The other custom footprints follow the same basic guidelines. My footprints are available on the **www.arrl.org/QEXfiles** web site. You can use them as a springboard for designing your own footprints.

Once you have annotated the schematic and assigned footprints to every device it's time to design the board.

Designing the Board

To design your board, start Pcbnew, the board layout editor that comes with Kicad. Do this by clicking the green circuit board icon in the schematic capture program.

The first order of business is defining the board edges. This is done on the Edge.Cuts layer. Before you start drawing the board, choose whether you want to use millimeter or inch units and set the Grid Size to 0.1 mm or 3.94 mils. I prefer to work in millimeters but switch to inches if a datasheet doesn't provide millimeter dimensions.

Next, locate a reference point on your drawing and zero the coordinate system to it. I like to start in the top-left corner of the board. Move the cursor to where you want to locate this corner then press the Space key. This will set the Relative Coordinates dx and dy to 0.0 and 0.0. Start your board from this location. Select the Add graphic line tool and draw a square to match the board size specified in the datasheet for your enclosure. Watch the dx and dy coordinates as you draw. This will help you make these edges exactly the right size. When you approach the starting location, a circle will appear around the line intersection when both ends match up. Click the mouse when the circle is visible to complete the board edge.

Next, locate the center of the board and place an alignment target there. Zero the coordinates to this location. This gives you a common reference for everything else on the board. Be forewarned that the board house may interpret this mark as a hole and drill a hole at this location so don't route any traces through this mark. Yeah, guess how I learned this lesson.

Once you've established the center of the board it's time to import footprints and the net list from the schematic capture program. This is done by clicking the Update PCB from the schematic button on the top button bar.

Once this has been done, all of the components will appear with white air wires connecting them. This is called the rats nest for good reason. Your next job is to sort this mess out! Initially all the components will be red, indicating they are assigned to the top copper layer of the board. To move them to the bottom layer, select each component and press the F key on the keyboard to flip the component. It will turn green to indicate it's assigned to the bottom copper layer. You can highlight and flip groups of components.

Mounting holes are treated just like any other component. Add them to your schematic then assign a footprint for them to define their size. In the PCB layout program, locate mounting holes first and position them exactly where the datasheet shows them using the dx and dy coordinates. Move a footprint by first clicking on it with the left mouse button to select it and then press the M (move) key on the keyboard. The component 'sticks' to the mouse cursor until you click the left mouse button again to release it.

Next, drag each major component to the location you determined in your initial scale drawing. Press the R key to rotate it to the left in 90° steps while you have it selected. Click the left mouse button to release it. Use the dx and dy coordinates to locate these components exactly where your scale drawing shows them. The quality of your finished project depends on these components being in the right place.

Take special care when designing a two board system so that components on one board don't interfere with the components on the other board. For example, on v1.0 of my controller boards, the pin header on the front panel was close to the top of the panel. I needed to use stacking 3.5 mm plug assemblies on the bottom board due to the number I/O connections I have. These stacked assemblies were tall enough that they interfered with the DuPont connectors plugged into the front panel's pin header. v1.1 of this board moved the pin header down 10 mm to eliminate this problem.

Pcbnew has a built-in 3D renderer that really helps visualize what your board will look like when it's finished. This is very helpful for finding many clearance problems.

I found that using two single-row SMT 0.1" headers worked well for my front panel board as the headers were located toward the top of the board and all the traces ran down the board from them. You can use a dual-row header if you can center it in the middle of the board and run traces both ways. Again, keep in mind that you don't want any traces crossing over other traces, and you need the ground plane to be unbroken so that every component referenced to ground has a path back to ground.

If you absolutely can't get away from having traces crossing over each other, add a 0 Ω SMT resistor to one of the lines and place this resistor on the board where it will bridge over the other trace. If you use wider traces (0.4 mm or more) you may need to use a larger SMT resistor to bridge the trace. I typically use 0603 sized SMT resistors and capacitors but go to 0805 size when I need to bridge a 0.4 mm trace.

Sometimes there is just no way around islanding a ground return as shown on **Figure 7**. You could use a couple of vias to connect the island to the front copper but this spoils the look of the panel. To get around this, add one or more 0 Ω resistors to your schematic and connect them all to ground as shown in **Figure 6**. These resistors allow you to build a bridge to your island. You can use vias in places where a control knob will hide them.

When you layout the PCB, position these resistors to bridge across the trace(s) that are blocking the ground path. R14 and R15 are circled in **Figure 7**. Your PCB layout software will probably complain about unconnected items because it doesn't understand the purpose of the resistors but the 0 Ω resistors will do their job and the LED will light. An alternate method would be to create a custom footprint for a jumper and just solder an insulated wire to each pad to form the bridge connection.

Producing the Board

Once you have the board designed, run the design rules check (DRC) to verify everything is connected and there are no clearance violations. Fix anything on the report. Next, go over the design several times manually checking the schematic and board to make sure everything is in order. It's easy to miss connecting a wire in the schematic capture program and the DRC in the PCB layout program won't catch it. Yes, another lesson learned!

Once you're confident you have it right, create the Gerber files for your board by clicking the Plotter icon. These files describe the physical layout of each layer of the board and tell the board house how to create your board. Again, there are several videos on the Internet describing this process and Kicad makes it nearly foolproof. You should have 12 Gerber files when you're done — don't forget to generate the drill files too. Zip these into a single file and upload them to the board house of your choosing. There are many Chinese board houses that provide good quality, low quantity hobby boards at very reasonable prices. I had **JLCPCB.com**

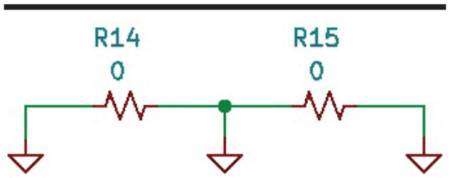

Figure 6 — Using 0 Ω resistors to connect isolated ground pads on a PCB.

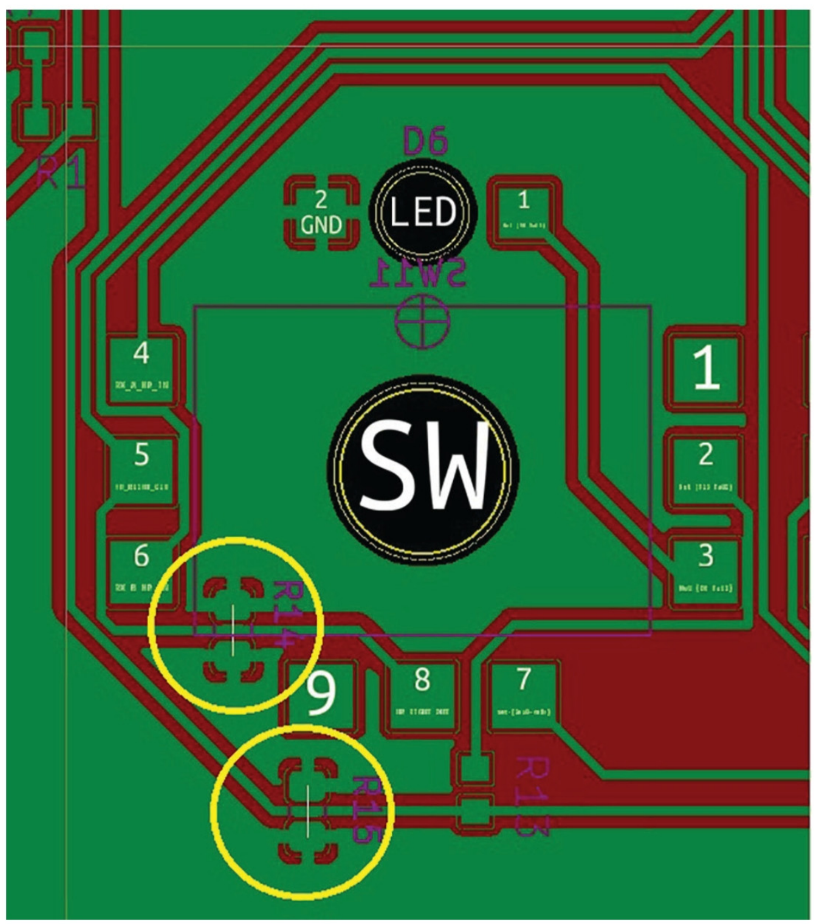

Figure 7 — Close up of two resistors connecting an isolated ground pad on a PCB.

create five of my front panel boards for $10 plus shipping. Check around with different board houses because some of them increase the price considerably if your board size exceeds 100 mm by 100 mm.

Wrapping it up

If you're lucky your new boards should show up in about two weeks and you'll be able to build a project you'll be pleased to display in your shack. **Figure 8** shows the completed front panel connected to the main controller board using 24 DuPont female/female jumpers.

Conclusion

I hope this article has given you some ideas on how you can create inexpensive, custom, professional looking front panels for your projects. If you haven't taken the plunge into circuit and board design perhaps this will inspire you to learn a new trick. Contrary to popular belief, old dogs can learn new tricks. After all, isn't that why you got interested in amateur radio in the first place? What other hobby offers lifelong learning opportunities like this one? I've been a ham for over 50 years and have never run out of things that spark my interest.

Lynn Hansen earned his Novice license WN7QYG in 1971, upgraded to General, WA7QYG, in 1972, and to Amateur Extra class, KU7Q, in 1981. Amateur radio provided a pathway into a career in electric utility communications where he spent almost 40 years, starting as a communications technician and eventually retiring as one of three operations managers over a seven state communication network. These days you can find him loving life in Lava Hot Springs, Idaho, and still learning and sharing ideas.

Figure 8 — Completed project showing the wiring interconnections between the front panel board and the main controller board.

[1] For more information on the SO2R controller and my other projects visit **https://ctr2.lynovation.com**.

[2] For more information on Kicad and to download the program, visit **https://www.kicad.org**.

[3] A via is a plated through hole in a PCB that connects the copper on the top to the copper on the bottom of the board. They are commonly used to pass a signal from one side of the board to the other or to connect ground planes together.

[4] The PacTec KEU-7 enclosure is available from Mouser and other suppliers.

[5] The PacTec KEU-7 datasheet can be found at **https://www.mouser.com/datasheet/2/314/drw_KEU-7-706759.pdf**.

[6] The data sheet for the pots I used can be found at **https://www.mouser.com/ProductDetail/313-1500F-10K**.