DIGWDF RenBuss User Guide to Construction & Configuration

Another strange product from DIGWDF – Dirknerkle's Inventorium and Generally Worthless Device Factory

Table of Contents

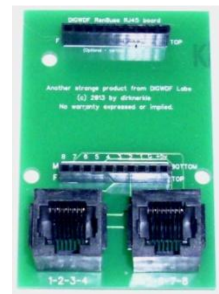
Overview

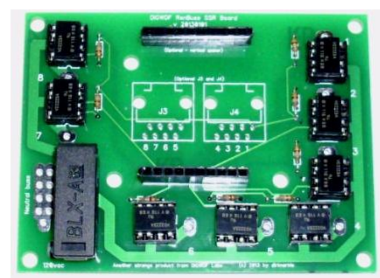
RenBuss is an inexpensive, flexible yet powerful light controller system that is easy to construct, configure and use. *RenBuss* comes from the blending of *Renard* and the *buss* wiring system where all controllers are connected together in a simple, buss wiring scheme. Simple soldering skill is required to assemble the RenBuss components, and theoretically up to twenty-three, 8-channel units can be connected together (184 total channels) using inexpensive wire. The system requires 4-wires from one unit to the next: two wires carry 6.3vac and power the controllers, one wire is the common data ground connection and the fourth carries the data control signals. RenBuss boards utilize piggyback connections and mixing and matching them gives the user considerable flexibility. The channels of the RenBuss SSR units are limited to 1 amp of current per channel and damage to the units may occur if this guideline is exceeded. For those needing more than 1 amp per channel, RenBuss is completely compatible with other DIYC external SSRs using the standard RJ45-type wiring convention. RenBuss can operate at any baud rate between 9600 and 115,200 baud; the kit is provided with PIC16F688 chips that are preset to 57,600 baud. (Wireless requires a maximum of 57,600 baud as well.)

RenBuss System Components

- **RenBuss Driver** – connects to the computer's serial port (RS-232 or RS-485) and an external power supply that can deliver adequate A/C current at 6.3vac. Current required: 125ma per RenBuss controller; 200ma per wireless RenBuss controller (use a separate wall wart for each wireless controller). The RenBuss driver provides both the signal and 6.3vac power for the controllers. A separate driver or RS-485 adapter is required for each RenBuss "network."

- **RenBuss Controller** – connects to the buss wiring system and decodes the channel signal. The controller is assigned an appropriate address (0 to 22) and has 8 active channels. Its companion board is either the RJ45 or the SSR board. Up to 23 controllers (184 channels) can reside on the same buss wire.

- **RenBuss RJ45 adapter** – used to route the controller's output channels to two RJ45 jacks; simple cat5 wiring can be used from them to remote SSRs. This is a handy feature for those who already have DIYC SSRs such as the SSREZ or DCSSR. Or, the adapter can be plugged into the RenBuss SSR, allowing the SSR to be connected to a controller via common, cat5 network wiring.

- **RenBuss SSR** – an 8-channel SSR (**S**olid **S**tate **R**elay) that translates the controller's channel commands directly to A/C controlled circuits. Maximum 1A per channel (max 5A total for all 8 channels of the SSR). The circuit board has space for two RJ45 jacks that can be used in place of using the RenBuss RJ45 adapter; this makes the RenBuss SSR completely compatible with any other DIY controller, connecting via cat5 network wiring!

- Optional add-ons:
  - **XBee Snap-In:** for wireless operation – replaces the wired buss with a wireless connection. Note that a wireless RenBuss controller requires a separate A/C wall wart or other power supply that can deliver 5 to 7vac at a minimum current of 200ma.
  - **MiniRenSI adapter:** provides a way to use a common RS232 or RS485 serial connection to the RenBuss controller from either the computer or another Renard controller such as a Ren64, SS16, or SS24. This replaces the DATA connection to the RenBuss controller with the common serial connection. This adapter requires the addition of a simple jumper wire patch to the bottom of the controller board.
  - **MiniRenSISO adapter:** provides full daisy chain capability from another Renard controller and then to the next controller using RS485 communication; replacing the DATA connection to the RenBuss controller. This adapter requires the addition of two simple jumper wire patches to the bottom of the controller board.

Tools You Will Need

We've designed RenBuss so that you won't need a lot of special tools. Most homeowners probably have most of the tools required to assemble a RenBuss system. You'll need:

- **Soldering iron**, typical pencil-type with a small, sharp point. A solder gun, the kind with the trigger, is not a good choice for electronic work. It's tip isn't sharp enough and it's much too hot – don't use one.
- **Solder for electronics.** Radio Shack is a good source – just a common 60/40 rosin core solder is fine. It's available in many diameters – the .032" diameter is a good universal one to use. Though it may be environmentally correct to use it, **WE DO NOT SUGGEST USING LEAD FREE SOLDER.** It is harder to use and requires not only a much hotter iron but greater soldering skill.
- **Wire cutter/ wire stripper.** One designed for electronics assembly is preferred as they can clip off excess leads closer to the circuit board, but many common wire cutters can work just fine. An old fingernail clipper actually works pretty well for clipping excess leads after soldering!
- **Small needle-nosed pliers or forceps.**
- **Small flat bladed screwdriver.**

General Construction Guidelines

Electronic projects are best assembled from the board upwards so that the first parts to be installed and soldered are the shortest components that often lay flat on the board: resistors, diodes, etc. Then start working your way upward with taller components (DIP sockets, female header sockets, header pins, terminal blocks) and finally the tallest components: RJ45 jacks, power wires (SSR), etc.
Some components are largely made of plastic (DIP sockets, female header sockets, terminal blocks) which are obviously sensitive to too much heat, so solder them appropriately. Other components such as crystal oscillators and capacitors can also be damaged by too much heat, so use enough to flow the solder but no more.

Some parts have polarity and must be installed in a specific orientation or they won't work. The PIC16F688 chips, ST485BN chips, VO2223A chips, diodes and electrolytic capacitors are examples. Another is the 18.432 mhz (megahertz) oscillator – it has a single square corner; be sure to align it properly matching the shape outlined on the controller circuit board. Before soldering terminal blocks ensure that their openings face the edge of the board, not the board's interior.

The circuit boards are clearly marked for the parts and their locations. After installing and soldering a part, be sure to clip off any excess leads from the bottom side of the circuit board. Take your time, use enough but not too much solder and build it right the first time. Step-by-step illustrated assembly instructions may be found later in this guide.

DIGWDF No-Fail Assembly Guarantee

If you purchased a RenBuss Kit from DIGWDF, you have our NO-FAIL assembly guarantee: if your board doesn't work after assembly, just send it to us and our engineers will fix it *FREE* – and you only pay for two-way shipping.

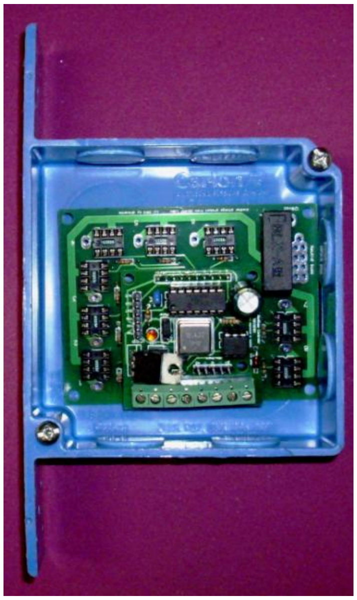
Supplies/Things You Will Need

The DIGWDF RenBuss Kit contains all the circuit boards and parts for the electronics, but you'll need to supply a few things on your own to make the system work. This is where a lot of DIY creativity comes in handy to keep costs down. Sometimes you'll find that a variety of solutions will work.

- **Buss wire.** This is the wire that connects the RenBuss controllers together and eventually to the RenBuss Driver. Four connections are needed, and the wire doesn't have to be fancy. Stranded wire will work better than solid copper wire because stranded is more flexible. Speaker wire works well, and so does common lamp cord. Cat5 wire isn't recommended because the internal wires are too thin; however, you could double-them up so they're in pairs – two orange, two green, two blue, two brown just twisted together at the ends. That would probably work okay for shorter distances. We've used 4-conductor speaker cable from www.monoprice.com (product ID# 4039) that's about $40 for 100' spool and what's nice is that the wires are color coded red, green, white and black -- makes connecting the buss very easy and trouble-free.
- **Extension cords.** Each RenBuss SSR requires 9 extension cords, believe it or not! One cord for each of the 8 output channels and one cord to power the SSR! You can often get some great deals at a Dollar Store, Costco, Target or other discount stores. Simple, two-wire extension cords with a plug on one end and usually a 3-outlet socket on the other end. Any length is fine – most DIY'ers use 6 or 9-foot lengths.
- **Enclosures.** Electronics don't do well when they get wet, and suitable enclosures are things you'll want to think about carefully. "Tupperware"-type food containers are generally popular, but remember that wires have to go in/out somehow. Office supply stores often carry some nifty and inexpensive boxes of various sizes – some from the Really Useful Box Co (yes, that's really a brand!). Check out www.reallyusefulbox.com and you'll see what we mean! Other good enclosures can be common plastic electrical outlet boxes from Home Depot or Lowe's. We've used a lot of Carlon blue PVC boxes (# A52151D), sealing the edges and holes with hot glue before we put them outside. Whatever you do, remember how gravity works: water runs downward. If the electronics are high in the box and the cables go out the bottom, it's a much better scenario than if the cables go out the side. For a limited time, DIGWDF is shipping 3 Carlon outlet boxes with covers in RenBuss Kits which many users have found handy for mounting small controllers and SSRs that have to be outdoors.
- **Computer.** A laptop or desktop is fine. Most DIY folks use a PC instead of a MAC because the free Vixen software (that's used to program and run the show) is designed for the PC. Running it on a MAC requires some tricky configuration, but many DIY folks have had success with Vixen on a MAC, too.
- **RS-232 cable**. You'll need to connect the RenBuss Driver to your computer, and this is the easiest way. Just an ordinary, 9-pin serial cable (not a null-modem cable) will do. If your computer doesn't have a serial port but has USB ports, a simple USB to Serial adapter (9-pin connector) is ok. You can use either USB to RS-232 or a USB to RS-485 adapter – either is fine.
- **Vixen software.** This is a free download from www.vixenlights.com . It's incredibly powerful stuff and has knock-your-socks-off capability. Really, it's absolutely free – K.C. Oaks, the author, won't even take any money for it!
- You'll want to become a member at www.doityourselfchristmas.com. We hang out there more than we do our own web site. It's the place to go for help in using Vixen, or making Vixen run on a Mac, or run on a Linux computer, or more. Membership is free, but a supporting membership costs only $20/year, and supporting members get access to group buys for electronic parts, kits, wire, connectors, lights both incandescent and LED, strobes, you name it.
- You'll probably need some tape (electrical and duct tape), probably some zip ties (cable ties), maybe some cable clips, nails, a hammer, pliers, etc. Typically these everyday supplies are used to route and mount cables up, over and around things on the outside of your house… or maybe on the inside if you plan to use RenBuss indoors instead.
- A piece of cardboard about 6" square. You'll use this during assembly.

Parts supplied with the RenBuss Kit from DIGWDF

- 10 circuit boards
    - 1 RenBuss driver
    - 3 RenBuss controllers
    - 3 RenBuss RJ45 adapters
    - 3 RenBuss SSRs
- All essential resistors, capacitors, diodes
- All header pins and sockets (you may have some leftover after your build)
- RJ45 jacks for driver and RJ45 adapter boards (but not optional extras)
- 3 Fuse holders and 5x20 5A fuses for SSR boards
- All essential IC chips and DIP sockets for the 10 included circuit boards
    - PIC16F688 chips are preflashed with firmware and labeled with addresses
    - Rectifier chips do not use DIP sockets and must be soldered-in
- 3 crystal oscillators (18.432mhz, half-can format)
- 4 rectifier chips
- 4 voltage regulators (5 volt, 1amp)
- A/C adapter (6.3vac, 1.5Amp wall-wart type – enough power for up to ten controllers)
- 3 Carlon plastic outlet boxes (model A52151) with covers may also be included in your kit. These provide excellent protection from the elements and a controller/SSR combo easily fits inside (see photo). There is ample room inside the box for wires and multiple punch-outs for cable exits. With just a couple zip ties (cable ties), it's easy to attach to a stake that's pounded into the ground or mounted onto the display itself, thus keeping the unit off the ground and the wires exiting downward out of the box, making the electronics inside quite well protected while also allowing ease of access out in the field.



Bill of Materials (numbers of parts are listed to build *one* of each component)

Part	RenBuss Driver	RenBuss Controller	RenBuss RJ45	RenBuss SSR
PIC16F688 microcontroller chip		1		
14-pin DIP socket		1		
8-pin DIP socket	1			8
6-pin DIP socket		1		
5mm terminal blocks	6	4		
ST485BN differential line driver chip	1			
L7805CV or L7805 5vdc regulator	1	1		
.1uf capacitor (non polarized)		1		
DF01 (or DF04) voltage rectifier	1	1		
47uf capacitor (16volt)	1			
3300uf electrolytic capacitor (16volt)	1	1		
4.3v zener diode	1			
9.1v zener diode	1			
5.1v zener diode		1		
1K ohm resistor (1/4 watt)	2			
680 ohm 1/8 or 1/4watt resistor		3		8
120 ohm resistor (1/4w)	1			
27K ohm resistor (1/4w)	2			
15K ohm resistor (1/4w)		1		
18.432mhz oscillator (half-can format)		1		
2-pin male header pin	2			
3-pin male header pin		1		
5-pin male header pins		1		
6-pin male header pins (ICSP)		1		
10-pin male header pins		1		
10-pin female header socket			2	2
RJ45 8-pin jack	1		2	(2 optional)
DE-9 female 9-pin PCB mount connector	1			
VO2223A optocoupler chip				8
5x20 PCB covered fuse holder				1
5x20 fuse, 5A max				1
6.3vac, 1.5A transformer (or equivalent)	1			
Buss hookup wire		X-feet (per your need)		
A/C power cord				9
Jumper Shunt	1	1		
Common 3 or 5mm LED		1		
RenBuss Circuit board	1	1	1	1

A complete listing of this "BOM" (bill of materials) including Mouser Electronics' part numbers is at the DIGWDF store:
http://diychristmas.org/store/product.php?id_product=75

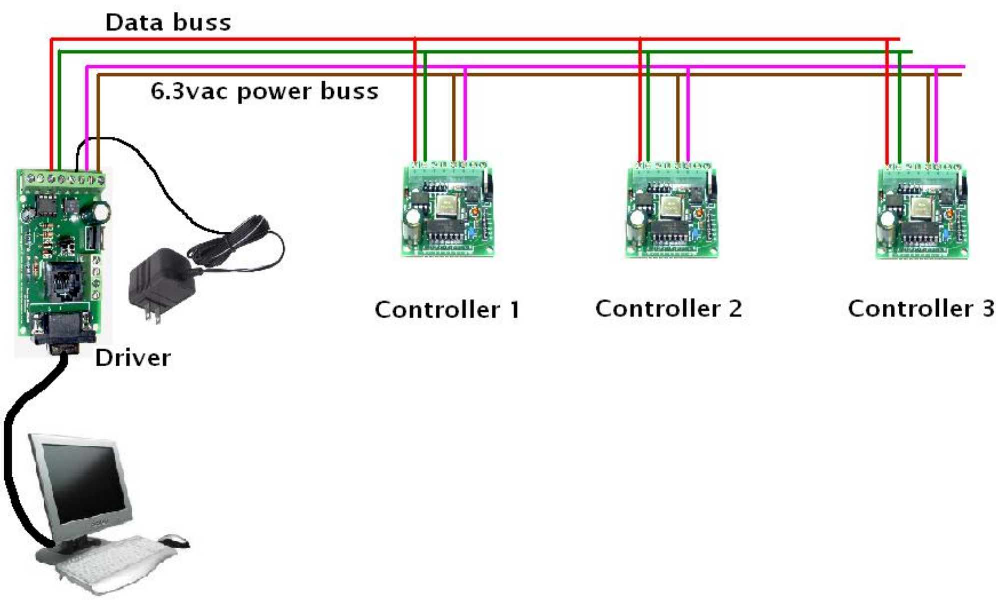
RenBuss Operation Concept

The 4-wire "buss" carries four things that are in common with each controller connected to the buss: 6.3vac (2 wires), data and ground. The 6.3vac power supply powers the controllers while the data and ground lines carry the control signals for all channels. Because each controller receives all channel data, each controller is given a specific channel address so that it will use only the desired 8 channels out of the total available channels (184 maximum). Each controller will always use a block of 8 channels out of the total; whether you have lights connected to those channels is up to you – you don't have to use all eight if you don't want to but the controller will always use 8 contiguous channels regardless of whether you have lights on them.

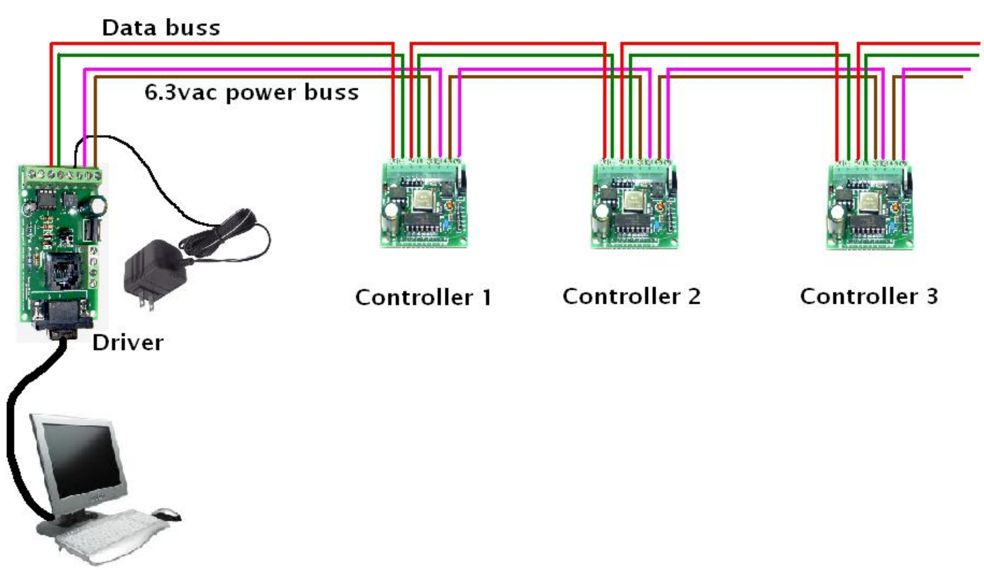
The controllers use standard, Renard start address firmware which is very versatile and provides outstanding dimming control of your lights. The firmware has several user-modifiable parameters, one of which is the address parameter. A handy chart has been included in this guide to help you determine the start address number to use for the desired block of 8 channels.

How does a “Buss” work?

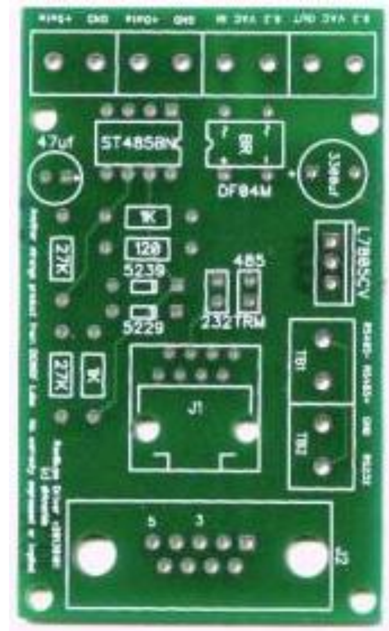
In the purest sense, a *buss* is a single continuous wire that goes from one end of an electrical system to the other end and all items that connect to it “tap” off that single, main line. RenBuss uses two busses of two wires each: the **6.3vac power buss** that powers all the controllers and a **data buss** that connects the computer signals.



However, it’s not always easy to run long lengths of single wires and tap off of them, and RenBuss’ terminal block connectors on each controller provide essentially the same continuous structure – it’s called ***“daisy-chaining”*** and it allows connecting another controller onto the end of the previous controller. This is more flexible and convenient than a pure, continuous wire buss design for it allows controllers to be added to the end of the line at anytime you want, or in between two other controllers simply by cutting the wires and plugging them into the appropriate IN/OUT terminal blocks. It’s also a lot fewer cables and less expensive than having to run a single cable from a centralized driver inside the house to each individual controller outside.



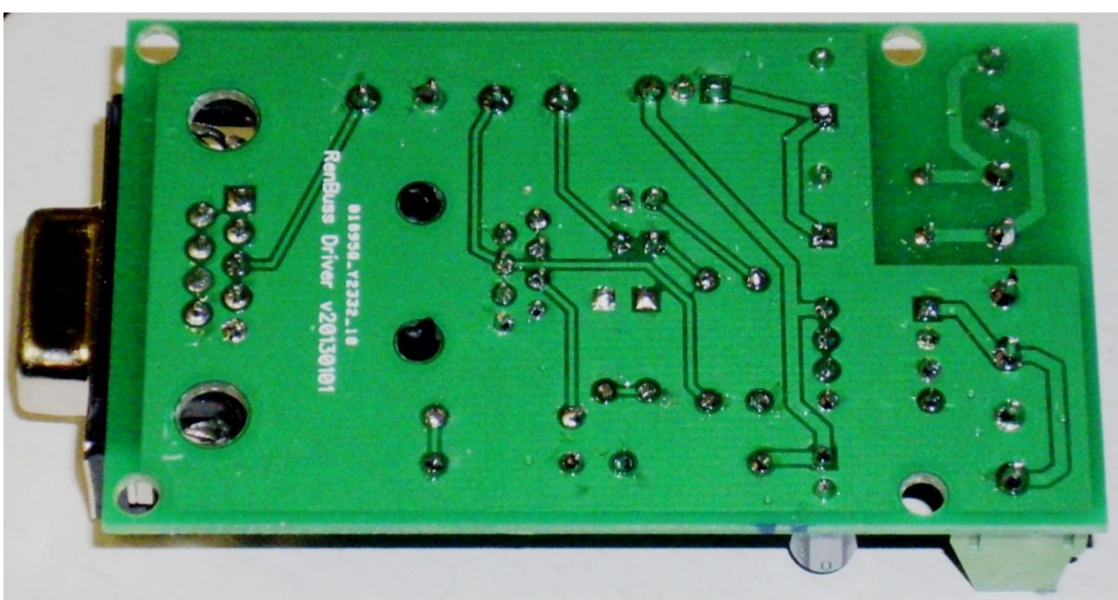
Step-by-Step Assembly – RenBuss Driver



1. Mount the 5229 and 5239 zener diodes (small glass parts each with a black band at one end) in their respective locations on the board. It's helpful to bend the leads with the needle nosed pliers so that the diodes are in a sort of upside-down U shape and fit into the holes. The parts are not interchangeable so be sure to mount the 5229 in the 5229 location and the 5239 in the 5239 location. Diodes are polarized parts so make sure that the black band on the parts matches the band on the board marking!
2. Lay a piece of cardboard down on top of the parts and hold them against the circuit board while you flip the board upside down. The cardboard keeps the parts from falling out when you flip it over. (Keep the cardboard handy – you'll use it whenever you need to flip the board over to solder parts in.)
3. Now solder the two diodes in place; afterward, clip off the extra wire leads. CAUTION: Don't cook these parts – they're pretty robust, but too much heat for too long can damage them.
4. Mount the two 27K resistors in the two locations marked 27K. It's helpful to bend the leads with the needle nosed pliers so that the resistors are in a sort of upside-down U shape. Resistors don't have a polarity to worry about but it's good electronic practice to install them so the colored bands read from left-to-right.
5. Mount the two 1K resistors in the two locations marked 1K.
6. Mount the 120 ohm resistor in the location marked 120.
7. Flip the board over and solder the resistors in place; afterward, clip off the extra wire leads.
8. Mount the DF01M (DF04 may be provided instead) bridge rectifier chip in the location marked BR. This is a polarized part so be sure to match up the **~ ~** markings on the part with the corresponding marks on the board; then the **+ -** markings will automatically match, too. A bridge rectifier converts A/C current into DC current, which the chips on the board need. Flip the board over and solder all four legs. CAUTION: Don't COOK this part. Use enough heat/solder to get the job done but don't overdo the soldering. Many parts won't need extra leads clipped off after soldering – like this one.
9. Mount a DIP-8 socket into the location marked ST485BN. Be sure to orient the notch in the socket so that it matches the notch on the outline on the board. Flip the board over and solder all 8 pins of the socket in.
10. The kit is supplied with a few strips of header pins – the strips look like small combs. These are easy to cut off with a wire cutter. Cut two, 2-pin headers from one of the strips. Insert these into the two rectangles marked "232" and the other with "485" at the top and "TRM" at the bottom. Flip the board over and solder these header pins in. These pins allow selecting the kind of serial signal that will be coming from your computer: either RS-232 or RS-485.
11. Mount the 47uf electrolytic capacitor in the location marked 47uf. This is a polarized part and the part itself is marked with a stripe on the side for the NEGATIVE pole; obviously, the other side is the positive side. On the board, the POSITIVE pole is the square one marked with the + so install the part with the poles in the proper positions. (Hint: the striped side is on the side closer to the edge of the board.) Flip the board over the solder the part in. Capacitors are quite robust but still, too much heat can damage them so don't overdo the soldering. Clip off the extra wire leads afterward.
12. Mount the 9-pin DE-9 connector onto the end of the board with the large rectangular box marked J2. Flip the board over and solder the part in. Afterward, you can use your needle nosed pliers to bend over the metal clasps that protrude from the larger holes on either side of the DE-9. They are not soldered-in.
13. Mount the row of four terminal blocks along the opposite end of the board from the DE-9 connector., where "6.3VAC OUT, 6.3VAC IN, GND, Data+, GND, Data+" is marked. Note: the terminal blocks may be provided in single form but they interconnect together. Be sure to connect them together into a row of

four blocks before soldering them in or they won't fit. Also make sure that the terminal block openings face out toward the edge of the board – this is where wires will connect. Flip the board over and solder the terminal blocks in.

14. Mount a row of two terminal blocks in the location marked TB1 and TB2, making sure the openings face out toward the edge of the board. Flip the board over and solder them in.
15. Mount the RJ45 jack into the center of the board marked J1. Carefully push the 8 pins through the holes and "rock" the plastic pins of the jack into the two alignment holes, then push down securely. Flip the board over and solder the jack in.
16. Mount the L7805CV (or L7805) voltage regulator into the location marked L7805CV. This is a polarized part: note the double-line on the board that stands for the metal tab on the part; the metal tab faces toward the edge of the board. When done, it's okay to bend it gently so it stands straight up.
17. Flip the board over and solder the part in. CAUTION: Voltage regulators are quite robust but too much heat for too long can damage them. Clip the excess leads off afterward.
18. Mount the 3300uf electrolytic capacitor in the location marked 3300uf. This is a polarized part and like the other capacitor you installed in step 11 above, the stripe on the side of the part denotes the NEGATIVE side; meaning that the + side is the other lead. Mount the part accordingly with the poles in the proper position. (Hint: the striped side is again on the side closer to the edge of the board).
19. Flip the board over and solder the capacitor in. Remember, electrolytic capacitors can be damaged by too much heat. Clip the leads off afterward.
20. Install the ST485BN chip into the DIP-8 socket. This is a polarized part so be sure that the notch on the end of the part matches the notch on the socket (now you know why you took care to install the socket so it matched the notch on the board!) Before installing the chip, you may need to bend the leads inward just a bit so they match up with the socket. They bend quite easily just squeezing them between your fingers. When you push the chip in, push it straight in and not sideways at all.
21. This completes assembly of the RenBuss driver. Flip the board over and check all your solder connections; they should be shiny and none of them should have large "balls" of solder on them. You may want to use a magnifying glass to check your work, making sure that there aren't any solder "bridges" (too much solder that connects two solder pads together or two parts together). If you find any, use a toothpick and heat the bridge with your soldering iron so it melts and use the toothpick to undo some of the solder. Of course, be careful not to COOK the parts or apply heat for too long on the solder pad. Too much heat on a pad can cause it to separate from the board, too, causing other problems. When done, the bottom solder-side of your completed Driver board should look like the one below:



Connecting the RenBuss Driver to Your Computer

The RenBuss driver provides three different ways of connecting to the computer:

- 9-pin serial cable (using the DE-9 connector and a standard serial cable)
- Cat5 cable (using the RJ45 jack and a custom-wired connector, more later…)
- Direct wire (using the two terminal blocks on the side of the driver; marked RS232, GND, RS485+ and RS485-)

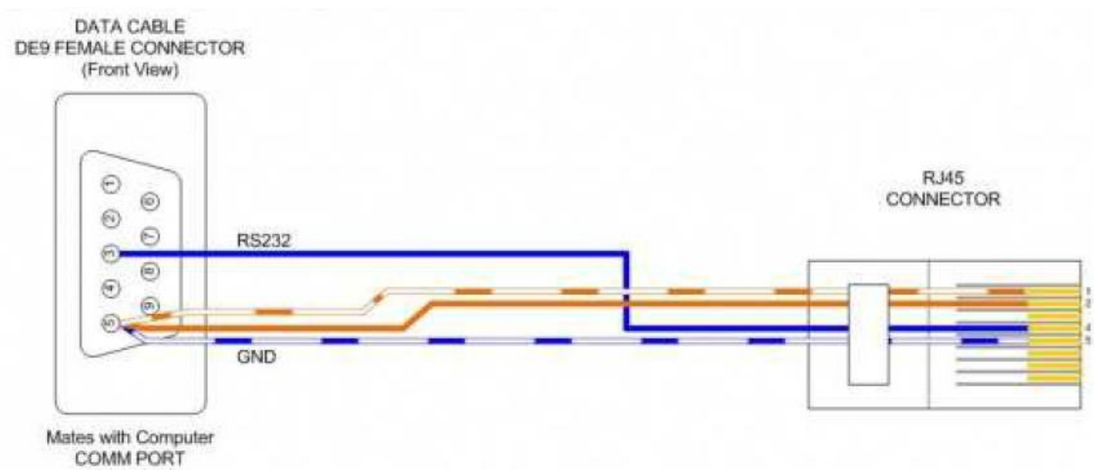
By far, the easiest method is to use a standard serial cable or USB to RS-232 adapter: just plug it into the driver and connect the shunt jumper on the pair of header pins marked “232.”

The next easiest connection is if using an USB to RS-485 adapter: the adapter probably has some outputs marked GND, D+ and D-, or possibly T+ and T- or at least some markings that indicate ground, the + signal and the – signal. You’ll need 3 wires: connect G of the adapter to the side terminal block marked GND; connect the adapter’s + line to the RS485+ side terminal block and the adapter’s – line to the RS485- side terminal block. Then connect the shunt jumper on the 485/TRM pair of header pins instead of the 232.

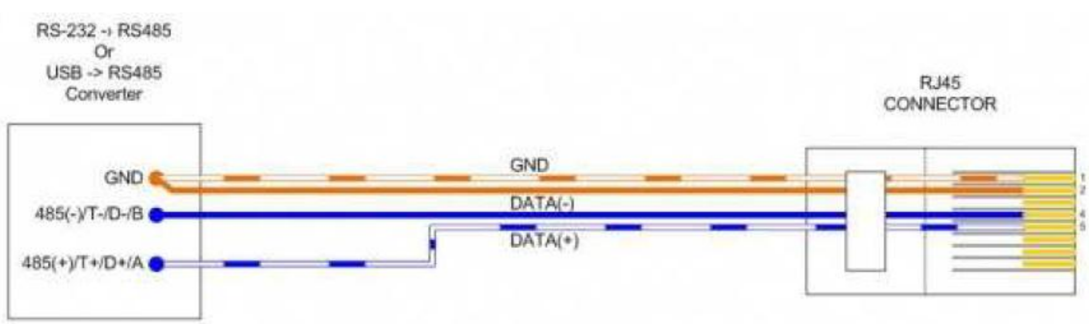
The third way is to use only the GND wire and wire from pin #3 of a standard RS-232 serial cable to connect to the side terminal block: connect the GND wire to the GND terminal block and the pin #3 wire to the RS232 terminal block. Then connect the shunt jumper on the pair of header pins marked “232.”

While convenient, easily the most confusing method is to use a custom cat5 cable and plug the RJ45 end into the RJ45 jack. Either RS-232 or RS-485 may be connected this way but most DIY enthusiasts have trouble making the cable because they’re not familiar with cat5 cabling and wire colors or RS-232 pin numbering. In the examples below, the RJ45 connector is positioned as if you’re looking at it through the BOTTOM – the bottom is the side with the metal connection pins:

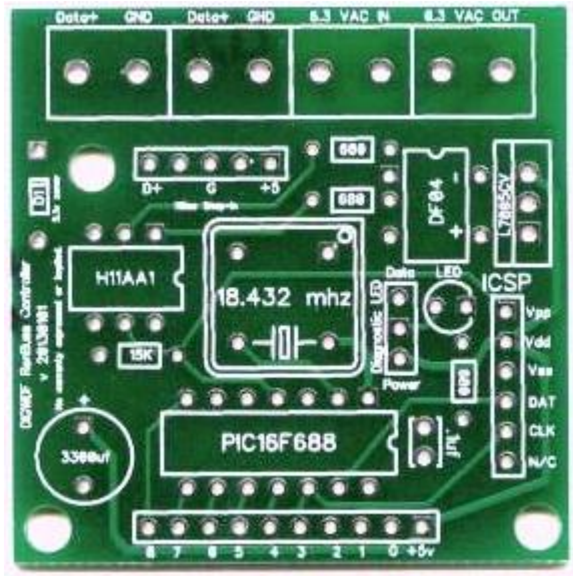
Wiring common RS-232 to an RJ45 connector is as follows:



Wiring RS-485 to an RJ45 connector is as follows:



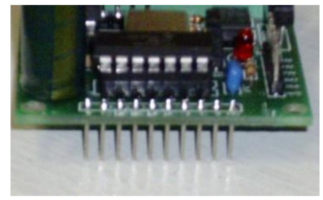
Step-by-Step Assembly – RenBuss Controller



1. Mount the three 680 ohm resistors in the locations marked "680". Flip the board over, solder them in and trim off the excess leads.
2. Mount and solder the 15K resistor in the location marked 15K. Trim off the excess wire.
3. Mount the 5.1volt zener diode in the location marked D1 (just left of the upper-left mounting hole). This is a polarized part and has a black stripe around one end of its glass body. Be sure to mount the part with the stripe matching the marking on the board (the strip is on the TOP toward the upper-left corner of the circuit board). Zener diodes are a little sensitive to heat so don't overdo the soldering. Of course, afterward, trim off the excess wire.
4. Mount the DF04 (or DF01) rectifier chip in the location marked DF04. This is a polarized part so be sure to orient the chip so that the + and – markings on the chip coincide with the markings on the board. Flip the board over and solder it in. No need to trim off any excess wire on this part.
5. Mount and solder the .1uf capacitor in the location marked .1uf. This is just to the right of the PIC16F688 chip location. This is not a polarized part and it doesn't matter which side goes in which hole. You may want to use your needle nosed pliers to straighten the leads so that the part fits all the way down flush with the surface of the board. Of course, trim off the excess leads after soldering.
6. Mount the small LED into the location marked LED. An LED is another form of diode, so it is not only a polarized part, it's also a bit sensitive to heat. Notice that one diode lead is shorter than the other; this is called the "cathode" and it fits in the square hole (on the right). Another way to tell which wire is the cathode is to look at the LED itself – the wire that connects to the larger part that looks like a flag is the cathode. Push the part all the way down to the board; flip the board over and solder it in. Clip off the excess wire. Note: it's not necessary that this part be pointed exactly upward, so don't try to bend it afterward or re-solder it to straighten it. Too much heat can "cook" an LED.

7. Mount the 6-pin DIP socket in the location marked H11AA1 and the 14-pin DIP socket in the location marked PIC16F688. Be sure to match the notches in the sockets with the markings on the board. Flip the board over and solder them both in.
8. Mount the crystal oscillator in the location marked 18.432mhz. This is a polarized part that has one corner that's just a little sharper than the other corners, and it may have a dot on the case top that denotes pin #1. The board marking shows pin #1 in the upper right corner, just below one of the 680 ohm resistors. Flip the board over and solder it in. Clip off the excess wires. A crystal oscillator is also a little heat-sensitive, so take it easy with the heat.
9. Cut a length of 5 pins from one of the header strips, a length of 3 pins and a length of 6 pins. You can cut a length of 10-pins as well, but set that one aside for now.
10. Mount the 5-pin header in the location marked D+ G +5 (just above the left of the crystal oscillator). Mount the 3-pin header in the location marked Diagnostic LED (just to the right of the oscillator), and the 6-pin header in the location marked ICSP. Flip the board over and solder them in. Hint: It's a good idea to solder only one pin on each for starters and then flip the board right-side up to check for straightness. It's a good idea that header pins are in straight for it makes it easier to use them later on. If one is crooked, hold a finger on the top side and lightly touch its bottom solder joint for a second with the iron to release it – allowing you to straighten it – it will get hot but the solder will set quickly and you'll have a nice, straight header afterward. Finish soldering the headers in place.

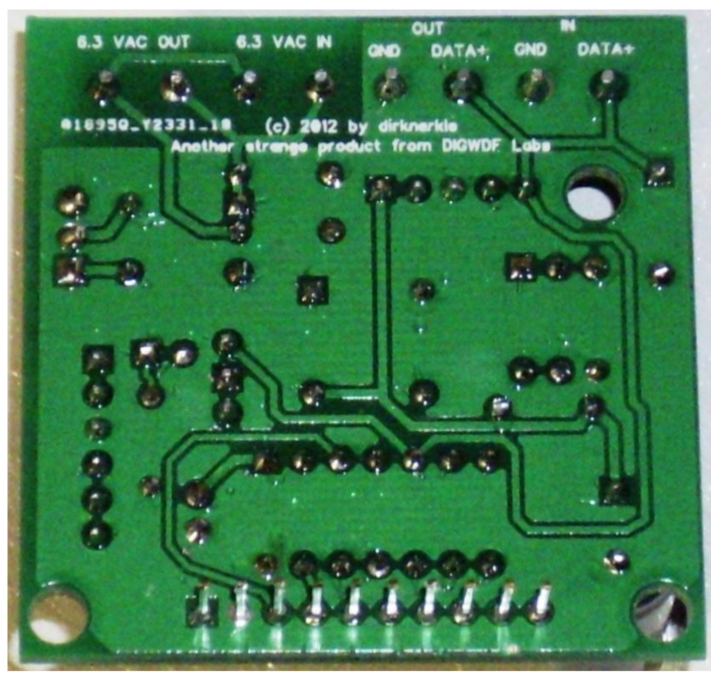
11. Mount and solder four terminal blocks into the locations along the top edge of the board. Be sure the holes are facing outward toward the edge of the board so wires can be connected later.
12. Mount and solder the L7805CV (or L7805) voltage regulator in the location marked L7805CV. This is a polarized part; note that the metal side of the regulator is on the INSIDE of the board this time. Be sure it's oriented properly, flip the board over and solder it in. Remember that a voltage regulator is pretty robust and can stand some heat, but don't go crazy. When done, it's okay to gently bend the regulator so it stands straight up.
13. Mount the 3300uf electrolytic capacitor in the location marked 3300uf. This is a polarized part and the stripe on the side of the part denotes the NEGATIVE side; meaning that the + side is the other lead. Mount the part accordingly with the poles in the proper position. (Hint: the striped side is toward the bottom edge of the board). Solder the part and trim off the extra wire leads.
14. Now take the 10-pin header that you made in step 9 above and insert it into the long row of holes at the bottom of the board, but insert it from the BOTTOM of the board upward so that the long pins point downward; you'll actually be soldering onto the TOP side of the board this time. Solder one pin and check it for straightness. This is one header that you'll want to be very straight; adjust if necessary using your finger and the soldering iron on that one pin. When it's okay, solder the rest of the pins.



15. Insert the H11AA1 chip into the 3-pin DIP socket, being sure to properly orient the chip in the socket. The chip may have a notch in it or it may have a tiny dot in the top corner instead. The dot marks pin #1. Pin #1 is always the top left pin of a chip, and if the chip had a notch, the notch would be the "top." So pin #1 should be in the location like the example to the right:
You may have to squeeze the chip's pins slightly with your fingers so they align with the socket properly; then push the chip straight in.

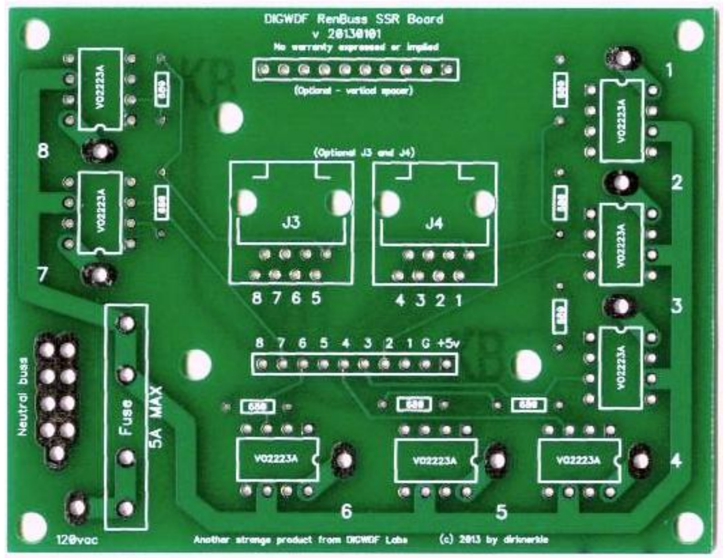
16. Insert the PIC16F688 chip into its socket. You may need to squeeze the chips pins for better alignment with the socket holes. When ready, push the chip straight in.

NOTE: Sometimes a pin on a chip will bend if it's not straight and in the socket when you push it in. Simply use your needle nosed pliers to *gently* grab both ends of the chip, rock it ever so slightly back and forth as you *gently* pull it out of the socket. Straighten the pins with the pliers (here's where a forceps comes in handy!) and then re-insert it into the socket.



17. This completes the assembly of a RenBuss controller. When done, the bottom of your board should look like this one, with nice, clean, shiny solder joints and a set of 10 pins sticking out the bottom. (Note in this example, the plastic collar around the header pins has been removed. This isn't necessary but is an option that can make for a better connection– just use a sharp knife to *gently* pry the plastic collar away from the circuit board and gradually work it up and off the header pins.)

Step-by-Step Assembly – RenBuss SSR



1. Mount and solder the eight 680 ohm resistors in the locations marked "680." Trim off the excess wire.
2. Mount and solder the eight DIP-8 sockets in the locations marked "VO2223A." Be sure to orient the sockets so that the notch in the socket matches the markings on the board.
3. Mount and solder the fuse holder in the location marked "Fuse." The fuse holder top is hinged and if you've already selected the enclosure you'll be putting the SSR in, you may find it easier to mount it with the hinge on the top or bottom – it doesn't matter electrically. After soldering, insert a 5A fuse into it and snap the cover shut.
4. *Don't insert the VO2223A chips into the sockets yet.* These chips are rather sensitive to heat and as the power wires need to be soldered to the board and those mounting holes are very close to the VO2223A chips. Soldering the power wires can require a little more soldering time – and heat – and to prevent damaging the chips, complete the soldering first and put the chips in last.
5. Now you can decide how you plan to use the SSR – whether you'll finish it in the RenBuss way using some of the female header connectors, or whether you'll connect the controller directly to it and solder it in permanently, or whether you'll install the two optional RJ45 jacks at the J3 and J4 positions. There isn't a right or wrong way, just lots of options. Here are some ideas for you to ponder…
    a. If it's your goal to permanently connect a controller to the SSR, that's easy – just insert the controller's 10-pin header into the row of holes marked +5v, G, 1-8, flip the board over and solder it in. Clip off the extra length of pins from the bottom when you're done. This effectively creates a controller that's very similar to the Renard SS8 – it's completely self contained.
    b. For the most flexibility, remember that the controller's row of 10 header pins coming out from the bottom of the controller board is intended to be plugged into a female header, making the controller removable (should the need arise) so you can perhaps plug it into an SSR board next year if you change your display layout. This is the RenBuss design – and the whole point of the male/female headers is that they provide flexibility. If you choose this option, simply mount and solder a row of the 10 female headers into the two narrow rows for them on the SSR board (easy to spot where they go) so that the headers point up from the TOP of the board. The row marked +5v, G, 1-8 is electrically connected to the channel outputs; the other row (at the top of the photo above) serves as a spacer to help hold the controller board secure; it's not electrically connected to anything.
    c. If you've connected the two female headers from step (b) above, then it's possible to assemble one of the RJ45 adapters so that it plugs into the SSR like the controller can, thus providing an RJ45 connection capability to the SSR and making it behave like any normal external DIY SSR. Then you'd connect the controller to another RJ45 adapter and connect the controller to the SSR via common, cat5 wire.
    d. Lastly, if you believe that you'll never want to plug the controller directly into the SSR, then you can certainly permanently install two RJ45 jacks into the J3 and J4 locations – and there'd be no reason to install the two rows of female headers on the SSR board at all. This would make the SSR a truly dedicated SSR unit much like the other popular external DIY SSRs, the DirkCheapSSR, SSREZ, SSRNEON, SSR8 and DCSSR, and you'd always connect it to a controller using a cat5 cable.

e. Lots of options for you to consider and they're all good. Your choice. And remember, nothing is really "permanent" in the DIY world. You can always un-solder something (although it takes a little practice and a couple extra tools such as solder wick or a vacuum-type desoldering iron).

6. Attach eight extension cords to the SSR *one at a time* as follows:

    a. Clip off the male extension cord plug, leaving about a foot of cord attached to it. This will allow you to possibly use it for something else later.

    b. Separate the two wires about for about another 6 inch length and strip off about ¼" from each wire.

    c. If you already have an enclosure for the SSR in mind that has exit holes for the cables, fish the cable through the enclosure hole from the outside to the inside before soldering it to the SSR – it's a lot easier to do this now than cut extra holes later.

    d. Notice that the insulation on one of the extension cord wires is smooth wire and the other one is either ribbed, has a rougher finish or possibly a line on it. The ribbed wire is (supposed to be) connected to the "neutral" pole of a two-wire circuit. On an extension cord this is the connection with the wider hole on the female end and at the plug end, this is the side of the plug connector with the wider blade. Solder the "neutral" wire to one of the holes in the SSR board called "Neutral Buss."

    e. Solder the other "smooth" wire to one of the channel holes marked with a number (1-8). This is called the "hot" side of the 2-wire cable. You can solder them onto the top or the bottom of the board, it doesn't really matter… although it might be best to push channels 5 and 6 wires up from the bottom in case you want to plug the RJ45 adapter into the top of the board – having the wires coming out from the bottom of the board may be helpful in that way.

    f. Label the female connector end of the cable with the channel number it's connected to. This will come in handy later when you connect lights to it!

7. Attach the 9th extension cord but this time, cut off the *female* end of the cord – leaving about a foot of cord attached to it so you can use it at a later time. (In the DIY world, nothing goes to waste!)

    a. Separate a couple inches of the wires and trim ¼" off the ends of each wire.

    b. Push the cable through the enclosure's exit hole like you did with the others, and solder the ribbed cable (the "neutral" wire) to the last, open hole of the "Neutral Buss" holes.

    c. Solder the smooth wire to the hole marked "120vac." This is called the "hot" side of the two-wire cable.

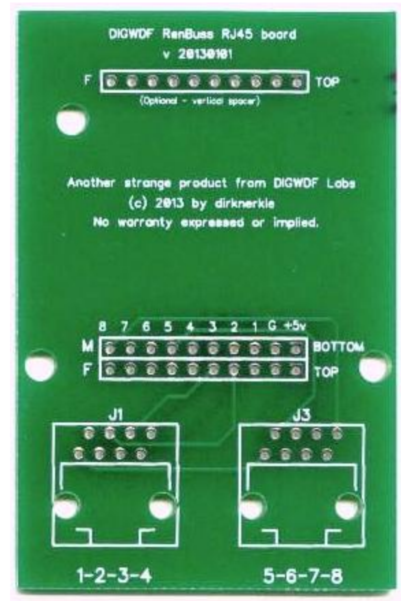
8. When all 9 cables have been connected, it's a good idea to bunch them together into a bundle and tie them with a zip tie (cable tie) on the INSIDE of the enclosure, just before it goes out the exit hole. When wires are bundled together like this, they effectively become stronger and less likely to pull out of a solder connection from accidentally tugging on wires when you put the unit out in the field. It's a good idea to put another zip tie around all the wires on the outside of the enclosure, right next to the exit hole.

9. Now insert the VO2223A chips into the sockets. You may want to squeeze the pins with your fingers so they align with the sockets better. You may notice that there are only 7 pins on the VO2223A chip yet the sockets have 8 places. Don't worry about it – one is unused. However, be sure to orient the tiny notch on the chip (next to pin #1) in the socket properly – the notch end of the chip corresponds to the notch markings on the board. The VO2223A chip's pins are easy to bend so when you push in the chip, push it straight in.

10. This completes the assembly of the RenBuss SSR.

You may wonder why the RenBuss SSR fuse is limited to 5 amps of current when the board has eight chips rated at 1 amp each. It's a safety issue. The power traces on the SSR are hefty and mirrored on both the top and bottom of the board thus doubling their current-carrying capacity, but even so, 8 amps is a lot of electricity to run through such a small board.

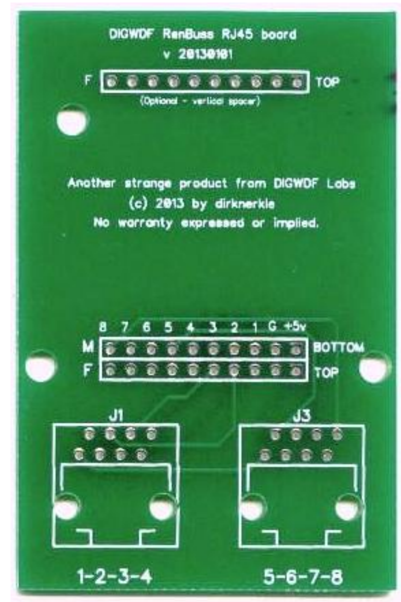
Step-by-Step Assembly – RenBuss RJ45 Adapter



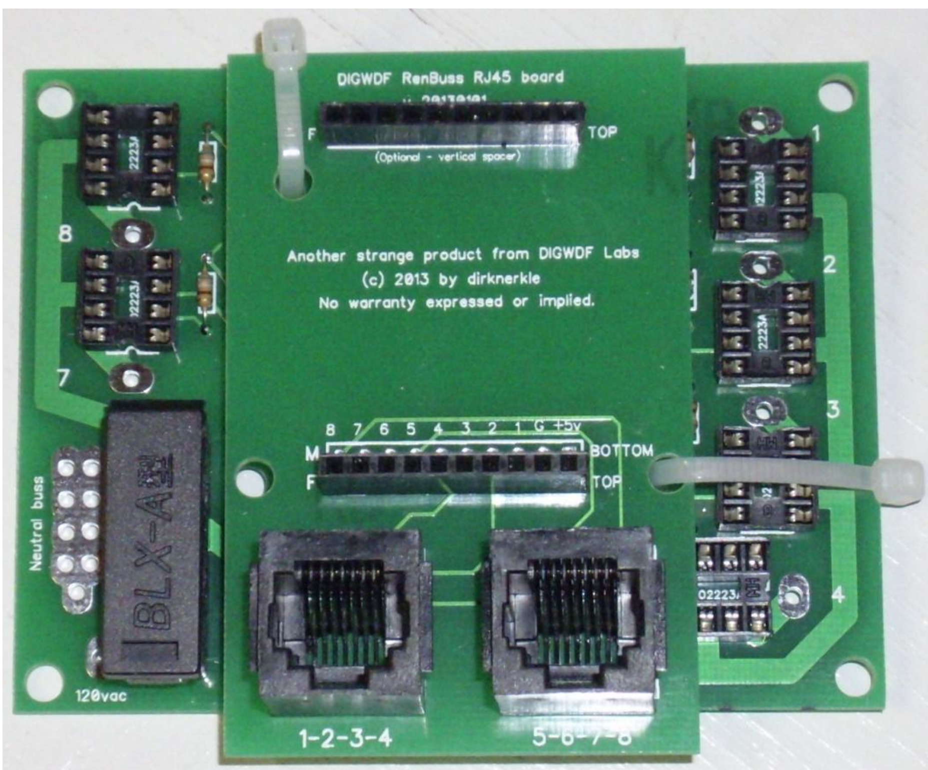
The RJ45 adapter may be used in the following ways, and depending on your plan, you may or may not have an immediate use for the RJ45 adapter.

- The RenBuss controller can plug into it, resulting in a way to connect common cat5 cables from the RenBuss controller/RJ45 to other common, DIY external SSRs. Examples of these are the wildly popular DirkCheapSSR, the SSREZ, SSRNEON, SSR8 or DCSSR. Or, you can even connect it to the RenBuss SSR (if you install the two optional RJ45 jacks in it). The advantage of using cat5 cables to connect the two is that your display piece can be a good distance away from the controller that's controlling it – for example, you may have the lights way up in a tree and the controller in a watertight box near the base of the tree.
- You can assemble the RJ45 adapter so that it plugs into the RenBuss SSR using header pins mounted onto the *bottom* of the RJ45 adapter much like you did with the RenBuss controller. Then you wouldn't install any RJ45 jacks on the RenBuss SSR board at all.
- You can assemble the RJ45 adapter and permanently solder the RenBuss controller to it. This essentially becomes a very compact, standard, Renard 8 controller for use with external SSRs only.

1. If you plan to plug a RenBuss controller into the RJ45 adapter:
   a. Mount and solder a 10-position female header into the row of holes at the top of the board marked "Optional – vertical spacer." This header only serves as a support to help hold the controller secure.
   b. Mount and solder another 10-position female header into the row of holes marked "TOP" at the far right side of the row. The "F" at the far left represents "female" header.
   c. Mount and install two RJ45 jacks at positions J1 and J3.
   d. You're done! Plug the RenBuss controller into the header that's closer to the RJ45 jacks. Your setup will then look like this:

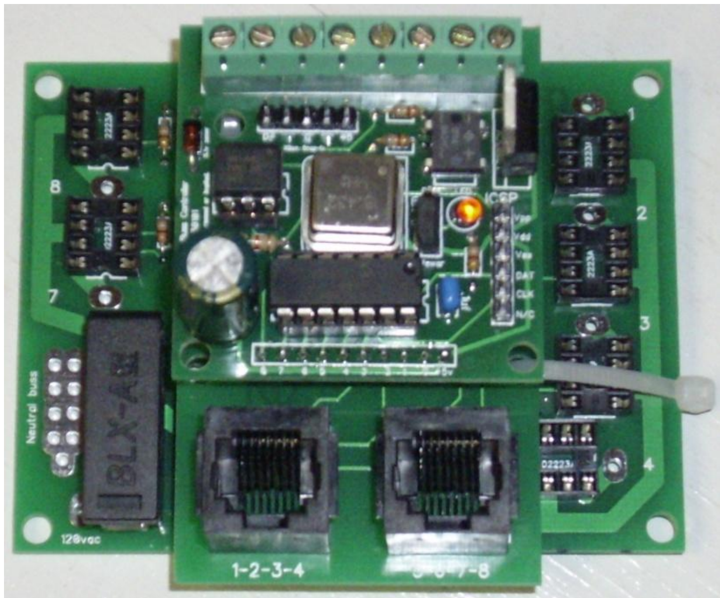


2. If you plan to plug the RJ45 adapter into a RenBuss SSR and connect to the SSR using cat5 cable:
    a. There's no need to solder the two female headers onto the top of the board (but you might as well anyway – it'll save time in the event you want to use it later).
    b. Mount and solder two RJ45 jacks in the positions marked J1 and J3.
    c. Cut a length of 10 pins from one of the header strips.
    d. Mount and solder the 10-pin header, but push it up from the bottom through the row of holes marked "BOTTOM" and "M" at the left side. "M" stands for male header pins. Solder one pin and then check the bottom for pin straightness – you want it straight so it plugs in easily. Adjust if necessary with a fingertip and a touch of the soldering iron on the one soldered pin; then finish soldering the rest of the pins.
    e. You're done! Plug the RJ45 adapter into the RenBuss SSR's female header that's closer to the front of the SSR – the row marked +5v, G, 1-8. Your setup will then look like this one (although the VO2223A chips had been removed prior to taking the photo):

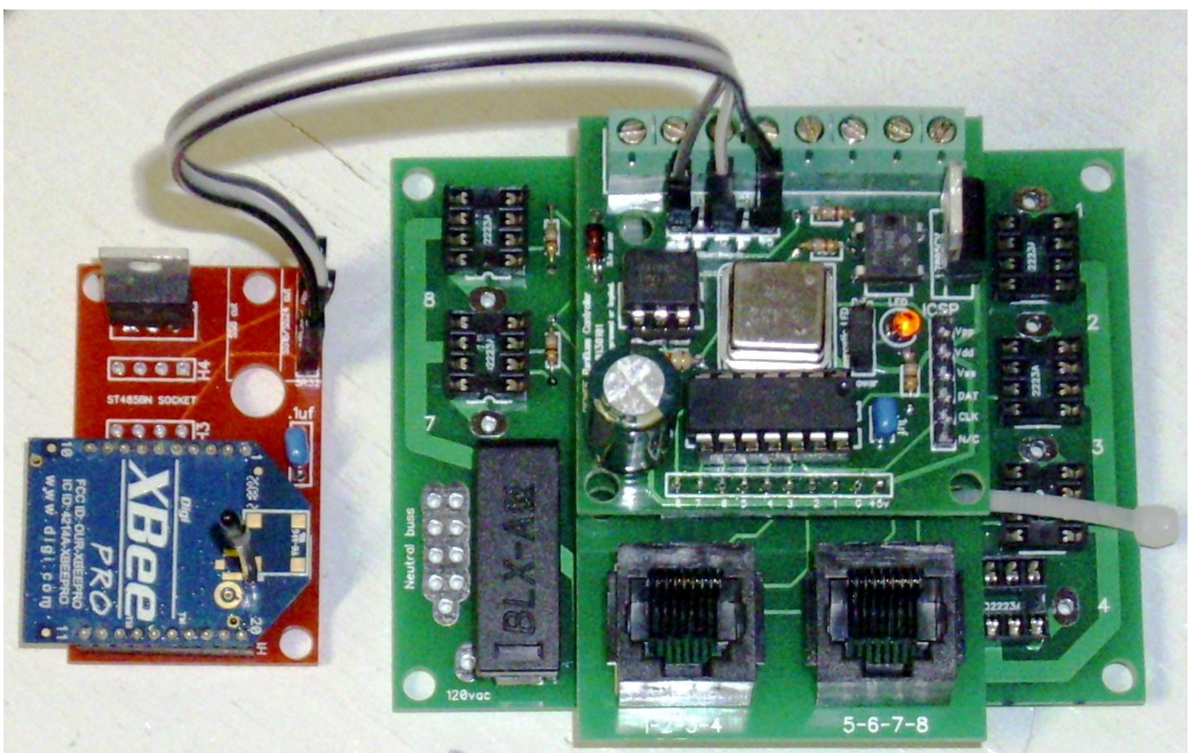


    f. Note that cat5 wire can be a bit stiff and it's a good idea to run at least one zip tie (cable tie) through the mounting holes of the RJ45 board and through the corresponding holes of the SSR, like the example above. This will keep the RJ45 board snugly connected should the cat5 cables be pulled. Plastic/nylon zip ties are preferred to wire or screws since they are not electrically conductive.

3. Can you plug all three boards together? Yes, you certainly can, and you get a RenBuss "Stack" like this:



When all three boards are interconnected, you have the best of all worlds because not only is the controller managing the 8 channels of the RenBuss SSR, you could connect cat5 cables to the RJ45 jacks and control external SSRs as well -- using the same channels, of course. While you're at it, consider adding an XBee Snap-In board and make the entire controller wireless. That all it needs is its own 6vac wall wart for power and you have a sort of "Super-Renard 8" like this:



RenBuss Construction Troubleshooting

Before you put your RenBuss system into operation, it's a good idea to take some voltage measurements first to make sure that it's constructed properly. To start, you'll need some diagnostic tools. One of the basic diagnostic tools is a voltmeter or DVM for "digital volt meter." Another term is VOM for "volt-ohm-meter." A DVM can usually do more things than an older VOM, but either can serve as a valuable diagnostic tool to measure whether the proper voltages are present where they're supposed to be.

But it may be best to begin by understanding what some of the electronic parts do because while RenBuss is intended to be quite a simple system, there's some pretty good technology going on behind the scenes. Understanding what some parts do can help you diagnose and fix problems.

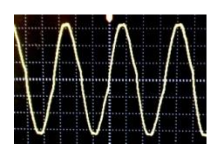
NOTE: electricity can be very strange – when there is no load on the circuit you'll find voltages are higher than what they'll be during actual operation. The main point of the tests outlined here is to determine whether there is ENOUGH voltage. For example, if you're expecting 5 or 6 volts but you measure only .25 volts, it means something's wrong. Likewise, if you're expecting 7 pr 8 but you get 15, that's a problem, too. But if you're in the ballpark with the values here, that's good enough for now.

- **DF01 (or DF04):** bridge rectifier. This converts A/C current into fully-rectified D/C current. Computer chips are almost universally powered by DC current and this single rectifier chip is an efficient way to do that for relatively small levels of current, such as only 1 amp. Using your DVM, on the **~ ~** side of the chip you should find approximately 7-7.3v of AC current using a transformer rated at 6.3vac. On the **+ -** side of the chip, you may find anywhere between 6.3-8.64 volts of DC. If the DC value isn't there, either there is a short in the board or the rectifier chip is damaged – *cut power immediately.*

- **L7805CV (or L7805):** This single chip limits the amount of voltage going through it to 5vdc and at up to 1 amp of current. The chips used by RenBuss (and most computers, by the way) are designed to operate between a certain voltage tolerance and 5vdc is the prescribed design spec for most of them. Higher voltage (in this cased 6-9 volts) comes into the regulator; the regulator uses about 1.5volts of it on its own, but then it lets through only 5 vdc to the rest of the parts on the board. Voltage over its rated output are essentially converted into heat and dissipated into the air by cooling. In RenBuss, the regulators won't get hot at all because the input voltage is fairly close to the output voltage. Each RenBuss controller also has its own L7805CV (or L7805) chip, too. Using your DVM, the CENTER pin and the metal back tab on the chip is "ground." Looking at the face of the L7805CV so you can see the printing on it, the left pin is the VIN (voltage in) pin and should measure about the same as the + - outputs from the rectifier chip – between 8-9vdc. The right pin should measure the regulated power, 4.9 - 5.04vdc. In fact, for voltage checking on the rest of the board, connect the ground lead of your DVM to the metal tab and leave it there because all ground points are connected together on a RenBuss board. Then you can easily check other voltage points using only the DVM's red + lead.

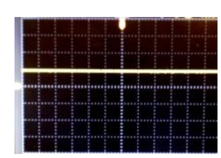
- **Electrolytic capacitors:** smooth out the electricity that powers the board which reduces electrical noise.

Larger values can store more electricity and therefore tend to be smoother still. You'll often find these near rectifiers and/or voltage regulators because A/C current has a "sine wave" shape to it – a positive and negative value that when converted to DC, creates only a + value and a zero (ground) value. However, with



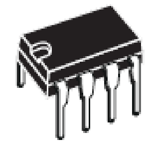
A/C power

DC current, imagine that the part of the wave that is below the center line is upside down and is only above the line – it's more like humps that go up and come back down to zero so that the value of the current still varies, but only to zero and not below it. The capacitors help fill in the gaps between the tops of the humps because they "store" electricity and give it out as it's needed. In the case of a power supply (rectifier/voltage regulator combination) it helps fill in where the voltage is not at the peak but is dropping toward zero, which lessens the humps and makes the voltage variance less prominent. This, in turn, results in cleaner, more constant power that the chips can use, which makes for better consistent performance. In fact, well-regulated DC current has no wave form at all – it's just a straight line on an oscilloscope:



DC power

- **ST485BN:** this is a popular RS-485 chip that converts computer signals into the proper TTL signals for transmission on the buss' data wire. The ST485BN chip requires 5vdc power and it should be found on pin #8. Since the output from this chip is connected to two terminal blocks, you could check the DATA+ terminal blocks – you should see between 4.9 and 5.0 volts DC.



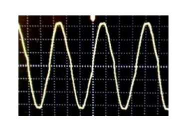
- **Zener diodes:** both limit the incoming + and - voltages to acceptable levels for the ST485BN chip. Has more application for RS-232 than RS-485 because RS-232's voltages tend to vary more. Zener diodes can also help prevent damage should serial input wires be accidentally reversed. No voltages to check here – just be certain the zener diodes are installed in the proper places and in the proper orientation. If either one is backwards, the circuit won't work. On the RenBuss controller, the 5.1v zener diode protects the PIC16F688 chip from the rare occasion that the data signal voltage is too high – the zener diode will limit it to 5.1volts, well within the acceptable range that the PIC16F688 can handle.

- **Resistors:** adjusts the electrical voltage/current to match the other parts. Electronic parts are designed to an electrical tolerance and resistors are used to help control the electricity flowing into them to either protect them from damage or to cause them to do certain things. No voltages to check here.

- **PIC16F688**: this is the computer part of the system. It's actually a complete computer on a chip, complete with all the parts that your desktop computer has only on a much smaller scale. While your computer has "gigabytes" of storage space, a PIC chip has only a few thousand bytes. And instead of using "software," it uses "firmware" because it is "burned into" the chip instead of loaded on the fly like when you use your laptop or desktop computer. The PIC16F688 does all the calculations and translates the channel data into electronic pulses that turn channels on and off. Connect the DVM's ground lead to the metal tab on the controller's voltage regulator and voltage at pin 1 of the PIC16F688 chip should be +5vdc.

- **18.432 mhz oscillator:** An oscillator creates a high-frequency timing signal which is used like a "clock" to keep everything operating together at just the right times. This is the most expensive single part of the RenBuss system and actually, you can eliminate it and use the internal oscillator built into the PIC16F688 chip instead. However, you are limited to running at 38,400 baud when using the PIC16F688's built-in internal oscillator and you must make a few changes to some of the settings in the firmware, after which you need to recompile the firmware and reflash the PIC chips.

- **H11AA1 chip:** this is called an "optocoupler" chip and it is connected to the 6.3vac input through two 680 ohm resistors (to limit the current a bit and protect the chip). This chip has some circuitry inside that can read when the A/C wave crosses zero voltage (the center horizontal line in the graph to the right). It sends a pulse to the PIC16F688 chip whenever it senses zero voltage (it's called a "zero cross" signal or ZC for short). The firmware in the PIC16F688 chip is looking for this ZC signal and it uses it to help determine how long to "turn a light on." Without getting into a long diatribe of what this means, if you flip a wall switch on and off really quickly, the light in the room doesn't seem to get to full brightness, does it? Well, in essence, that's how dimming works: if the light is only on for half the time, then it should only get to about half the brightness that it would if it was on all the time. The PIC chip runs very fast (18.432mhz) and can easily determine various lengths of time when the light can be on vs. off. But to do that, it needs to know when the power is actually at the zero point. The pulse signal looks like this:



ZC-pulse

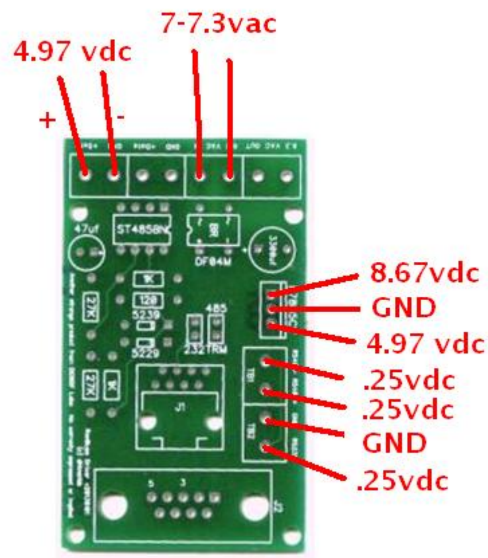
- **LED & 680 ohm resistor:** on the RenBuss controller, these (and the 3-pin header and jumper shunt) are actually optional parts and only necessary if you wish to put the diagnostic LED on the controller board. By connecting the jumper shunt across the top 2 pins the LED will flicker as data is received proving a good data connection; connecting the shunt across the bottom 3 pins will cause the LED to glow brightly proving a good power connection. The resistor lowers the electrical current so the LED won't burn out. Having a diagnostic LED is a convenience but is not necessary for the operation of the controller. Note that the application of the jumper shunt on the data line is for diagnostic purposes only and should not be connected during normal operation.

Voltage Measurement Test Points

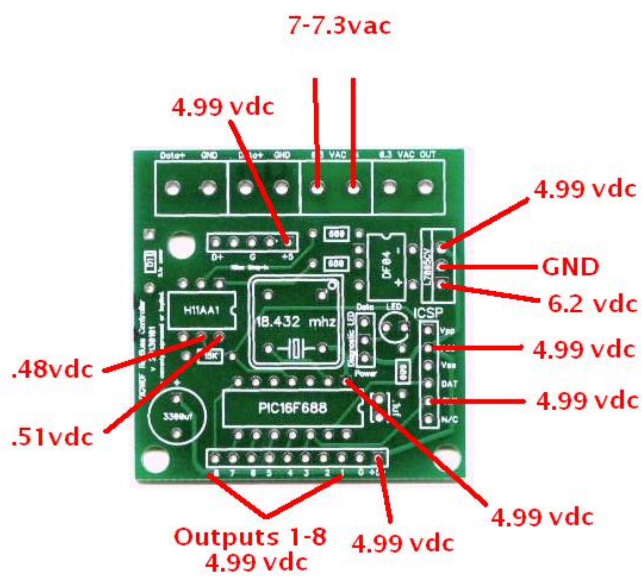
These are sample measurements and meant to be guidelines; your voltages may vary slightly but probably not very much. For convenience and accuracy, we're used the bare circuit boards to mark the voltage locations – you would, of course, test the *completed* boards. Note that ALL measurements should be made with the board powered up but without a load connected to either the 6.3VAC OUT terminals or to the +DATA/GND terminals. Also understand that the values you get may not be an *exact* match to those printed here – all that's important is that they're *relatively close* – within a half a volt or so.

A/C Measurements: be sure to switch your DVM into A/C voltage mode before testing. Measurement is taken across the two 6.3VAC IN terminals. You can also test across the **~ ~** terminals of the DF01M (or DF04M) chip.

DC Measurements: be sure to switch your DVM into DC voltage mode before testing. Measurements are with the DVM's GROUND (black) wire connected to the back metal tab of the L7508CV (or L7805) voltage regulator and testing the other points as necessary. On the RenBuss Driver, measure the voltage across the +Data and the GND terminal block, too. On the RenBuss Controller, the voltage of each of the 8 output pins should be identical with the board powered up.



RenBuss Driver



RenBuss Controller

A Logical Approach to Troubleshooting during Operation

The most common thing that will happen is that the lights connected to a specific controller won't come on. There are many things that can go wrong, and the best way to solve the problem is to start at the source and work your way all the way to the lights. But first, here are a few of the most common, obvious things to review. You'd be amazed how frequently the solution is something as simple as "I forgot to plug it in." Reminder: in a "buss" configuration, losing a connection somewhere in the chain of controllers usually affects all the controllers downstream from it. Therefore if controllers #1-2 power up but #3 doesn't, there's likely a connection problem of some kind between controllers 2 and 3.

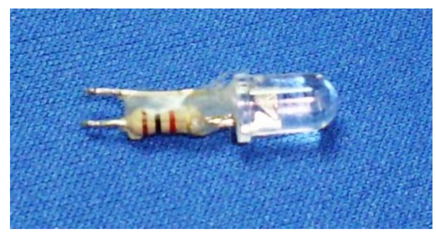
- Are the lights powered up now? If so, **UNPLUG EVERYTHING FROM A/C POWER FIRST!**
- Did the lights work before you plugged them into the controller? Did you test them first? If not, don't expect them to work afterward.

- Are the lights plugged into the proper channel of the SSR? (i.e. if the lights are plugged into channel 3 of the SSR is that channel also "on" in the Vixen software, too?)

- Is the SSR plugged into 120vac power? (SSR's are not powered by the buss; only the controllers are.)
- Is the 120vac power cable alive -- i.e. is the power on?
- Is the RenBuss driver appropriately connected to the power from the RenBuss Driver? (The "Diagnostic LED" header is just to the left of the LED on a RenBuss controller. Put a shunt jumper on the bottom two pins to test for power to the controller. If power is adequate, the LED will light brightly.)
- Are the power wires secure in their terminal blocks or are they loose?
- Are the data wires secure in their terminal blocks or are they loose?
- Is the designated channel (or channels) turned "on" in the Vixen sequence?
- Is the controller's address set properly so that it represents the designated channel?
- Is there a break inside the wire somewhere (which can happen if you step on it)? Use your DVM to test continuity from one end to the other with the wire disconnected at both ends.

If all the physical connections are correct, and assuming that the controller's address is set properly, you can check the following during run time:

- Put a shunt jumper on the <u>top</u> two pins of the "Diagnostic LED" headers (picture above). When data is flowing on the buss, the diagnostic led should flicker.
- If you didn't install the diagnostic header and LED, you can make a quick data continuity tester by soldering a 680 to 1K ohm resistor to one of the legs of an LED. In the example to the right, a 1K ohm resistor was soldered to the + leg of the LED, so I always know which leg of the tester is positive and which is negative. I used a little hot glue to hold the two leads apart so they wouldn't short. This tester can be used up to a 12v signal without harm and I've found it very helpful to have in my pocket when I go outside. To test during run time, just connect it across the +DATA and GND terminals. It should flicker with the data coming through. If there's no data, it should be solid on. Of course, you can also use it to test the +5vdc and G lines, too, although using a DVM tells you what the voltage is…



- If the proper voltage and a signal is getting to the controller, it's likely that the address of the controller is not right for the channel you're testing. Double check that the controller you're working with should really have the channels you think it does. Reflash the chip with the proper firmware and try again if not.
- If you're absolutely positive the controller's address is correct, use your DVM to test the voltage between the +5v pin and G along the long row of header pins next to the PIC16F688 chip. It should read very close to 5 vdc. Then check the voltage between the +5v pin and the channel that's supposed to be ON. If it measures zero or near zero, then the channel is OFF. If it measures 5vdc (or thereabouts) then the channel is ON and the problem is with the SSR or the connection to it. (Note: when a channel is 100% "on" the channel *pin* should switch to become a GND pin and so measuring across the channel pin and the +5 pin on the far right of the row should give a near 5vdc reading.
- If NONE of the controllers are working, check the communication settings for the Vixen plug-in you're using; it's most likely that they are not set to the proper BAUD rate. (Default: 57,600 baud). Vixen must be sending the data at the speed the controllers are set to receive it or RenBuss won't work.
- If the SSR isn't working, unplug the SSR from 120vac power first; then check the SSR's fuse.
- Unplug and reseat the controller in the SSR slot, and then repower the SSR with 120vac.
- Double-check that the light string plugged into the channel is in working order; the plugs on most light strings have fuses of their own and if one is blown, the string may not work.

Configuring the Start Address for a RenBuss controller

RenBuss actually use TWO busses – a power buss and a data buss. The power buss does just that – its 6.3volt of A/C electricity powers every controller connected to it. The data signal is traveling along a different pair of wires called the data buss and is picked up at each controller, too. However, if each controller used the same address, each controller would react to the same channels. And if you have a 24-channel RenBuss system (3, 8-channel controllers) you likely want one of them to take the first 8 channels (1-8), another to take the second 8 (9-16) and the third to take the last 8 (17-24). That's why the PIC16F688 chip on each controller is flashed with a different address stored in its firmware. The PIC16F688 chips in your RenBuss kit are flashed with addresses 0, 1 and 2 (see the chart below) and each chip's address is marked with a sticker that starts with the address and is followed by the channel numbers it will use: 0/1-8, 1/9-16, 2/17-24 (etc.).

Do I have to use a different start address in each PIC chip? Yes, if you want each chip to decode a different range of channels. But if you wanted two or more of your controllers to have the same address, that's perfectly fine – then they will react to the same channel commands.

Does the first controller have to be the lowest address, and then going upward with each controller? No, it doesn't matter which controller has which address because all controllers will receive all the channel data. It only matters that the controller has the right address for the controls you want it to use!

Can I set a controller to use only 4 channels instead of 8? No, the PIC16F688 chip is an 8-bit chip and will always gather 8 channels of data. However, you don't have to plug lights into All 8 channels if you don't want to – if you only want to use 4, that's fine and it won't hurt the controller or any of the equipment at all.

What if I need a different address than what is provided in the kit? Contact DIGWDF and we can either reprogram your PIC16F688 chips for you or you can order a new set with different addresses. Or, you can purchase your own "PIC Programmer" and reprogram them yourself. The firmware is a free download from the DIGWDF store and is already pre-compiled and ready to load into a programmer and flash the chip. If you're ordering a kit and already know what addresses you need, just let us know with your order and we'll program them for you before we put them in the kit.

FYI, the formula for calculating the start address is easy: **A = (C-1)/8** where A is the start address to use and C is the first channel number that the controller will use. The formula produces the following chart:

Channels	Address to use	Channels	Address to use	Channels	Address to use
1-8	0	65-72	8	129-136	16
9-16	1	73-80	9	137-144	17
17-24	2	81-88	10	145-152	18
25-32	3	89-96	11	153-160	19
33-40	4	97-104	12	161-168	20
41-48	5	105-112	13	169-176	21
49-56	6	113-120	14	177-184	22
57-64	7	121-128	15		

Simply plug the channel number into the C value and perform the calculation. Note also that the value used for C must be evenly divisible by 8. For example, addresses 40 and 48 are valid but 44 is not.

Why do the channels stop at 184? This is calculated based on running at 57,600 baud and powering the controllers using a 3A, 6.3vac transformer at the RenBuss Driver board. The real issue is available current from the driver board to power the controllers through the buss wire, and 22 is about the maximum number you can have with 3A of current. If you used a pair of SPT-2 wires as your buss, it could be a lot more, but you'd also need a much bigger and more expensive transformer at the driver board, too! Note: RenBuss Kits include a 6.3vac wall wart that delivers 1.5A of current; enough for ten controllers on the buss line.

Interconnecting RenBuss Component Boards

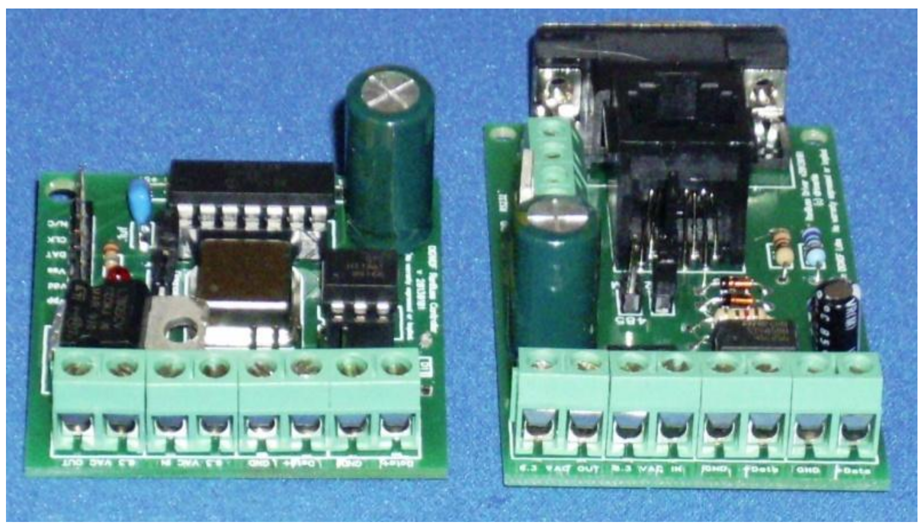
A RenBuss controller by itself needs either an RJ45 board or an SSR to connect to the outside world. The boards are designed so that the controller can simply plug into either the RJ45 or SSR board; the RJ45 board is designed for users who already have normal DIYC SSRs and want to use RenBuss with them. It incorporates two female headers, one for the electrical connection to the controller and the other simply for support. It also has two RJ45 connectors that have the pin out connections for common SSRs. Plugging a RenBuss controller into the RJ45 board makes for a very compact (1-7/8" x 2-3/4") 8-channel controller for external DIYC SSRs such as the SSRez, DCSSR, SSR8 or DirkCheapSSR.

Plugging the controller into the RenBuss SSR is just as easy. The SSR has two female headers, one of which is used for the electrical connection for the controller and the other simply as a support with no electrical connection at all, same as the RJ45 board.

The RJ45 board can be constructed with a set of male header pins installed on the bottom instead of the normal female header socket. It can then be plugged into the SSR board and used as a common, 8-channel SSR with other DIYC controllers such as the Simple32, Simple16, Simple8 or Ren64.

The RenBuss controller itself includes a 5-pin header where an XBee Snap-In board can be plugged in, immediately adding wireless capability to the board and replacing the wired buss with a wireless buss. Of course, in so doing, make certain that the baud rate of the XBee radio matches that of the RenBuss controller and that the RenBuss controller is powered by a suitable 5vdc wall wart.

Interconnecting the RenBuss Driver and Controllers

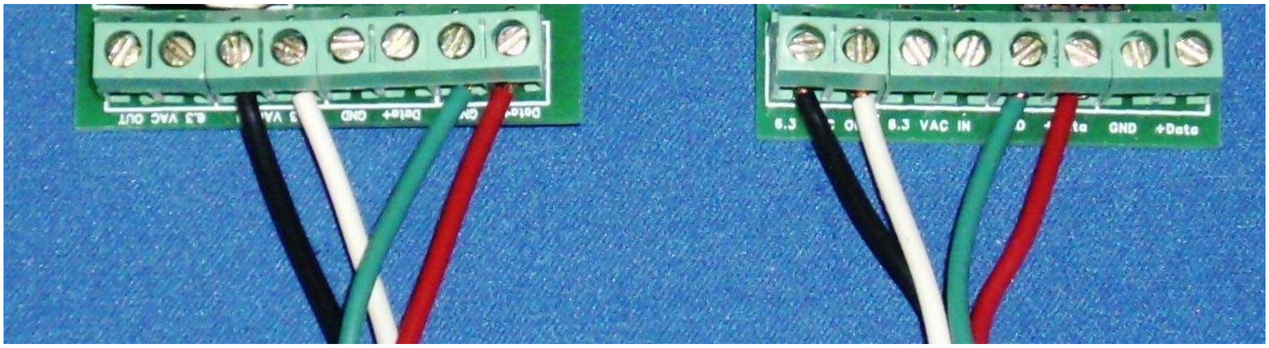


Analogy: an electrical "buss" is an electrical circuit inside your house where many outlets are connected together and protected by the same circuit breaker in the main power panel. All of the outlets get electricity at the same time, regardless of where they are in the house. When you plug too many lights, toasters, televisions and computers into the same circuit, the circuit breaker trips and all units lose their power.

The RenBuss is much like that, except that when there are too many controllers on the buss wire there won't be enough electricity left in the data signal to power all of them; it will appear as if a some sort of RenBuss circuit breaker has blown because they all will stop working. But they can also stop working by improperly connecting them together!

The RenBuss controller and driver boards each have an end with a set of terminal blocks. From left-to-right, each pair of connections is labeled as follows: 6.3 VAC OUT, 6.3 VAC IN, GND and +DATA, GND and +DATA.

The example below uses a 4-wire cable with colored wires for clarity. The CONTROLLER board is on the left; the DRIVER board on the right. The color of wires used is not important, but what is important is to notice that the 6.3VAC wires (black/white) are kept in the same left-to-right orientation from the 6.3VAC OUT from the DRIVER on the right side of the photo below to the 6.3VAC IN at the CONTROLLER on the left side. Also



notice that the green/red GND and +DATA wires are kept in the same orientation, too -- the GREEN connects to GND and the RED to DATA+. Bottom line: keep the wires in pairs and connect like terminals, such as connecting GND to GND or +DATA to +DATA; connect the 6.3VAC OUT from one unit to the 6.3VAC IN terminals at the next unit, keeping the polarity of the wires in the same left-to-right orientation.

Remember that the color of the wire is not important – *but what it's connected to is!* (Actually, the 6.3VAC IN and 6.3VAC OUT connections are tied together electrically so it doesn't really matter which is IN and which is OUT, but they're labeled that way to help you know which wire is going where – it's always helpful out in the field if you ever have to fix something!)

Screw-down terminal blocks were chosen as the connection method for RenBuss because they can accept almost any kind of wire and they are oriented for PAIRS of wires. So in short:

- **Never connect components when the A/C power is on. Unplug from all A/C power first!**
- **Always keep each pair of wires together**
- **Keep the wires of a pair in the same left-to-right orientation**
- **Never connect a 6.3VAC wire to ANY of the connections marked GND or +DATA! Doing so will cause damage to the RenBuss electronics.**

Connection Wire selection

A major concern is that if the connection wires are too tiny they will constrict the voltage and/or current through them, and this always results in wires getting warm and when they get too warm, they catch fire. Further, consider that 20 controllers at potentially 125ma each totals 2.5 amps, so the power wires carrying the 6.3VAC electricity should be of sufficient gauge so as to handle that amount of current. For example, either cat3 or cat5 wire (both inexpensive and usually 24 gauge) seems adequate for the task but each has a significant voltage drop at 100 feet and neither is designed to carry that much current. A better choice would be 18-gauge SPT-1 wire, 20-gauge speaker wire or wire used for low voltage garden lighting. The examples above used an 18 gauge, 4-conductor speaker wire from www.monoprice.com (PID# 4043).

Questions/Answers

Q: Why did DIGWDF develop RenBuss?
A: Our goal was to create a Christmas lighting control system that was as inexpensive and simple as we thought it could be made yet retain virtually all of the wonderful capabilities the Renard system provides. Currently, a popular, commercially-available lighting system with only 16 channels costs $260. A 16-channel RenBuss system (two 8-channel controllers with SSRs) can be assembled for about $60, saving $200.

Q: What's a "channel?"
A: A channel is the smallest number of lights that can be turned on or off together. You might think of a single string of 100 lights as an individual channel, which means that an 8-channel controller can control 8 strings of 100 lights – 800 lights! Or perhaps you might put two strings of 100 lights each together and call that a channel. If they're LED lights, you might put 3 or 4 strings together. A single 8-channel controller, then, could perhaps control upwards of 32 strings of LED lights. That's 3,200 LED lights on a single controller! However, most users probably have the most fun by putting just one string of lights on a channel so they can control each string of lights independently from all the others. This gives you the opportunity to put a string of red and a string of green on a bush and control each separately – or together if you like. With just a pair of controllers, you'd be able to change the colors to all red, or all green anytime you wanted, and if you turned them all on together, red + green gives off a sort of golden glow. A channel can also be a single bulb if you want – like lighting up Rudolph's red nose on a wire-frame mockup of a reindeer.

Q: I've seen some displays where the lights fade up and down in brightness. Can RenBuss do that?
A: Yes. You can illuminate the light intensity anywhere between 0-100% for any channel independently from all the other channels.

Q: What skills and equipment will I need to build a RenBuss system?
A: You'll need common hand tools such as a needle nose pliers, small screwdriver, wire cutter, pocket knife, small soldering iron and solder, inexpensive wire, and a PIC programmer. Radio Shack has almost everything you'd need although they don't carry many of the individual parts. If you already have a trigger-type soldering "gun," pick up a small soldering iron instead. The "gun" type gets much too hot for circuit board assembly. Another tool that can be really handy is a DVM – digital volt meter (or VOM, volt-ohm-meter). These are used for measuring electricity. You don't need an expensive one – a $6-$10 unit will be just fine.

Q: I don't know how to solder. Where can I get help?
A: Radio Shack has inexpensive learning kits and there are many "how to solder" videos on the web. It's very easy to do. Check out www.youtube.com and search for "how to solder electronics."

Q: I don't know one electronic part from another. Where can I get help with that?
A: The web, of course! Google "**electronic part identification**" and you'll find plenty of information. One of the best is: http://www.uchobby.com/index.php/2007/07/15/identifying-electronic-components

Q: If Radio Shack doesn't have all the parts, where can I buy them?
A: We buy most of our parts from Mouser Electronics (www.mouser.com). You can easily purchase on-line, and they even have a toll-free number where you can get help finding parts you need -- their catalog is huge with literally millions of parts. Other excellent sources are www.digikey.com and www.alliedelec.com. The DIGWDF store has a "universal BOM" (bill of materials) list in PDF or Excel format that outlines all the parts and Mouser part numbers for every DIGWDF product. The lists are free for the taking.

Q: Electricity scares me. How will I know I've built them right and safely?
A: The old joke goes like this: "Easy, by the amount of smoke you get when you plug them in!" Actually, most of RenBuss uses very low voltage A/C or DC electricity. Some of RenBuss does plug into 120vac, the same voltage that you use throughout your house. And yes, it can be dangerous *or lethal* if not handled properly. All of the RenBuss assembly is performed when nothing is plugged in, so you're certainly safe there. The wall wart transformer and the SSRs are the only parts of RenBuss that touch 120vac power and if you're worried about it, you can certainly send the assembled SSR board to us along with proper return postage. We'll test

and even repair it if it's not assembled properly, and that service is free. Or you could take some close-up photos of both the top and bottom of the board and email them to us. If the photos are clear, we can usually tell whether something's out of place and we'll let you know. Lastly, if you know someone who is comfortable working with normal house wiring, they're a very good sounding board. They may not know a lot about electronics, but they'll be able to tell by looking at the assembled SSR if there are any glaring problems.

Q: What kind of software do I need to run RenBuss?
A: A fantastic and readily available freeware program named "Vixen" is your best bet: www.vixenlights.com. Vixen is not available in stores and can't be purchased. The fellow who made it wanted it to be free to Christmas light enthusiasts everywhere and it's a wonderfully powerful piece of software that runs on all versions of Windows as well as many versions of Linux.

Q: Will I ever outgrow RenBuss and have to start over?
A: People who get into this hobby usually do expand their systems over time. And while RenBuss is expandable, it's only moderately so, and a full-fledged Renard system may be in your future. However, the electronic parts that make up RenBuss are, by design, *exactly* the same parts that the "big" Renard controllers use, so you won't be left high and dry.

Q: Will the RenBuss run DMX?
A: Yes. While the connection mechanism between controllers is not the same as DMX uses it can work just fine although you'd only use the driver board to provide electrical power to the controllers and the DMX signal would come from a DMX device. RenBuss uses the same DMX controller addressing system that Renard DMX uses and more information can be gleaned from the wiki documentation at doityourselfchristmas.com.

Q: Will RenBuss run at 115,200 baud?
A: Yes, but possibly only for short-length buss configurations; it may be too fast for setups where the buss connection is physically longer..

Q: Can I use a 10vdc or 12vdc power supply instead of 6.3vac wall wart?
A: No. The RenBuss controllers require low-voltage A/C power and 6.3vac was chosen out of safety concerns. But as each controller has a rectifier chip and voltage regulator, you could get by with a 9-12vac wall wart, too. However, the system is not designed for high-amperage, high-voltage A/C current. Using higher voltage than that is generally more dangerous and will likely damage one or more components.

Q: Can a wall wart be used at each wired RenBuss controller instead of powering it remotely?
A: Yes, then only a 2-line connection wire is needed for Ground and +Data. Common SPT-1 or SPT-2 lamp cord can be a handy and inexpensive solution. But it must be an A/C wall wart, not one that converts A/C into DC. Note that an individual A/C wall wart is required for each RenBuss unit that is to be wireless, too. <u>Important:</u> wall warts (small power supplies you plug into the wall) are notorious for providing inaccurate voltages. Be sure to measure the unit's output with a DVM to make sure it doesn't exceed 12 vac. Be sure the wall wart you choose can provide *at least* 150ma of current at an acceptable (6.3-12vac) voltage.

Q: Can I configure more than one controller with the same address?
A: Yes. Each RenBuss controller has its own address which has no effect whatsoever on any of the other controllers on the buss. You could set controllers 1, 7, 14 and 19 to use the same address and they would all react to the same channels simultaneously.

Q: Can I set a controller to start with channel 20 or 39 or any other arbitrary channel?
A: No. Because the PIC16F688 is an 8-bit chip and the Renard firmware for it is designed to manage 8 channels at a time, addresses must reflect 8-channel boundaries. The formula A=(C-1)/8 must result in a whole number (an integer) for the address to be valid. Plugging 20 into the formula: A=(20-1)/8 comes out to 2.375 which can't work. The address must be either a 2 or 3, not a decimal number in between. HOWEVER… there's no law of blinky-flashy that says you must connect lights to all channels, so in this example, if you set the controller to address 2, it would use channels 17-24; you could plug lights into 20-24, leave 17-18-19 unused and it would have the same net effect.

Q: Where can I get the firmware for the PIC chip?
A: A link to download the pre-assembled HEX code is at the DIGWDF store as a download for the RenBuss kit. The download is free as a ZIP file. After downloading to your computer, simply double-click on the downloaded file to open it, choose the address you want based by the file name and drag it out of the folder to your desktop. The DIGWDF store is at http://www.diychristmas.org/store.

Q: How do I program the chip with an address?
A: You'll need a PIC Programmer such as the PicKit-1, PicKit-2 or PicKit-3, all readily available from Microchip directly (www.microchip.com) or elsewhere on the Internet such as eBay. Beware of PicKit "clones" as their quality and compatibility varies widely. Be sure the unit you select can handle PIC16F688 microcontrollers. Once you have the programmer plugged into your computer, you simply load the HEX file for the address you need (see previous question), insert the chip in the programmer and "write" it. It only takes a few seconds.

Q: Can I have the chips programmed for me?
A: Yes. Contact DIGWDF – we can do that for you free. You would buy the chips, send them to us with a list of the addresses you need and speed you want to use (either 38,400 or 57,600 baud), include return postage and we'll return them to you. Each will be marked with its respective address and speed (ex: 11-57600 would be address 11 at 57,600 baud; 4-38400 would be address 4 at 38,400 baud).

Q: How far apart can the controllers be?
A: The data line is probably the more limiting factor here than the 6.3vac power lines, and the real maximum is unknown. We successfully tested RenBuss up to 150 feet and being that RenBuss was designed with the smaller installation in mind; we weren't really concerned with "maximum wired distance." If the RenBuss controllers are wireless, the distance between units can be hundreds of yards, just like the bigger Renard systems that use the same XBee Snap-In board.

Q: Will a 16 or 24-channel version of RenBuss be developed?
A: There is no plan to expand beyond the granularity of small, compact, 8-channel controllers because of our overall design goal which is to keep the system inexpensive. However, you could easily put two or three RenBuss controllers inside the same enclosure and accomplish it that way! Otherwise, there are plenty of 16, 24, 32, 48 and 64-channel units available for those who want or need high-channel count controllers.

Q: Can I control motors and fluorescent lights with RenBuss?
A: No. Such devices usually require more start-up current than the RenBuss SSR can manage. In addition, fluorescent lights are generally not dimmable, so using RenBuss in either instance is not advised.

Q: Is RenBuss compatible with other DIY controllers?
A: Yes and no. Most Renard controllers are operated at 57,600 baud, and so does RenBuss. However, it's possible to run RenBuss at 38,400 and eliminate each controller's crystal oscillator and related expense. To be compatible, all controllers must be running the same speed. Secondly, RenBuss uses a different physical interconnection method than normal Renard controllers, so it's a bit harder to connect the two. However, if RenBuss controllers are configured as wireless using the XBee Snap-In board option, it's 100% compatible with other wireless Renard controllers as long as all are set to run at the same speed, or you may add a MiniRenSI or MinirenSISO adapter to each RenBuss controller which makes it plug-in compatible with virtually every other Renard controller.

Q: I see there are holes in the RenBuss boards. What are they for?
A: All RenBuss component boards have mounting holes that line up vertically. For a truly secure installation, it may be helpful to lightly bolt the boards together using small screws and appropriate nuts. Another method is to use a pair of long, thin zip ties to loop down, under the bottom board and up again, finally linking together on the top.

Q: I like the RenBuss concept; is there any way to accomplish the same thing with my existing SS controllers?
A: Yes. You need Y-connector type RJ45s for each of your SS controllers. These are often sold as 8-connector adapters for telephone systems, are plastic and are designed to plug into Rj45 jacks. Plug the Y-connector into the RS-IN port of the SS controller. Then plug the RS485 cable into one side of the Y-connector. And instead of using the RS-OUT port on the SS controller, you connect the outbound cable to the other side of the Y-connector, and then run it to the next controller. In this way, all controllers get the same channel data simultaneously, and if the first PIC in each SS controller is set with its own start address, it will then react only to those channels.

Q: If I can do it with my existing controllers using an inexpensive Y-connector, why in the world would I buy the RenBuss system?
A: Good question. *You* wouldn't. RenBuss was not designed for you; it was designed for newcomers to the lighting hobby whose budgets and needs are less and quite possibly whose electronics kit construction skills are undeveloped, too. RenBuss uses far fewer parts than most all other systems, requires fewer tools and in much less sophisticated than DMX and Ethernet-based systems, yet, it provides significant bang for the buck. It's intended to be "the next step" from those who might currently be using X-10 gear to simply turn their outside lights on and off. RenBuss gives them a very powerful system with full dimming control that is still entry-level difficult to implement. And, as users' needs and skills grow (as we all know they will), RenBuss component parts can easily be reused with other Renard equipment.

Q: If I run into trouble, is help available?
A: Yes. Contact us via the DIGWDF Store or use the contact link at www.diychristmas.org.

Q: How much current does a string of lights use?
A: A typical string of 100 mini-incandescent lights will use about 1/3 of an ampere of current. As you'd expect, a string of 50 mini-incandescent lights is about half that. The electrical current usage is usually printed on the box itself and may be listed as a decimal number and abbreviated, such as .39A or .16A. A string of 100 LED lights takes a fraction of that required by incandescent lights – possibly .06A *or even less.* So while each channel of the RenBuss SSR may be able to handle 2 strings of 100-count mini-incandescent lights (16 total for the SSR), because LEDs require such little electricity, each channel may accommodate *ten* strings of LEDs (80 strings total for the SSR).

There is a wonderful and inexpensive (about $25) electrical current management tool called the **Kill-A-Watt.** In a real-time display, it can display the voltage and HZ (frequency) of your A/C power, the current being used (when something is plugged into the unit's front outlet), or convert the current into watts. It's easy to measure a string of lights – just plug it in and press the AMPs (or watts) button. Then just multiply that value times the number of strings on the circuit and you'll have an accurate measure.

While we're on the topic of electrical current, let's consider ***current reserve***. You home has many electrical circuits, and you probably have a circuit breaker box; older homes may have a fuse box. Each circuit is designed for a maximum number of amps of current. Typical home circuits are 15 or 20 amps. But you should never max-out the current on a circuit – you should always keep about 20% in reserve. So on a 15 amp circuit, expect to use no more than about 12 amps; on a 20 amp circuit, that means use no more than 16 amps.

Why is current reserve important? Well, sometimes certain devices such as a refrigerator need a little extra current when the motor "starts up" so it can regenerate its refrigerant and keep the food cool. If your circuit is maxxed-out with lights, there's no current reserve left for the refrigerator's start up and it will likely trip the circuit breaker when it tries. If you're not home to monitor it, you won't realize the refrigerator's circuit isn't on anymore and the food inside is going bad by the minute without you knowing it. Another example might be a sump pump; without a current reserve, it may blow the circuit breaker when it starts up, resulting in a flooded basement if you're not there to address the situation.

Saving Money

One of the reasons people go the DIY way is to reduce cost. The estimated pricing in this guide is more-or-less related to the retail costs of the parts; shopping around for better prices is part of the fun, and there are many, many places to find excellent value. The last page of this guide contains a list of common parts that are in many DIY lighting projects, and while sometimes you may need only one or two of a certain thing, sometimes buying in quantity saves a lot – and then you have a supply of parts on hand so you can jump into yet another project or even repair a circuit should that become necessary. For example, buying a couple dozen resistors might cost $2.40 but a pack of 100 of them is about $3. Many of the parts supply companies are overseas and our experience has been very positive with those listed; delivery is generally fairly quick; some of the China-based companies take an extra week or so. Another great source is eBay -- for RenBuss testing and development, we bought a 250 foot roll of 18-gauge, shielded, 3-conductor wire (used for sprinkler and/or security systems) for only $50 including delivery.

Plan ahead. Ordering a single part is extremely costly – the shipping charges can eat you alive. You can also post a note on one of the forums at www.doityourselfchristmas.com that you're looking for a certain part and more than likely, several people will offer to send one to you. Before you order something online, take stock of what parts you already have and those with low quantities and wait an extra day. Buying on impulse over the Internet is usually not an economical way to shop.

Learn to take your time assembling electronic projects, and you'll save money, too. You'll want to finish it quickly and immediately plug it in and see how it works. This is usually not a very good idea. You need to check and double-check every part you've soldered in, checking for good solder joints, checking for solder bridges which are usually accidental and which connect two parts that aren't supposed to be connected. Make sure parts are installed correctly with the right orientation – remember, many parts are polarized and only work one way. Another tip is this: *don't get ahead of yourself*. Most quality DIY electronic projects have step-by-step assembly instructions: do a step and actually make a check mark over it so you know where you are in the project. Don't skip steps unless they don't apply to your assembly – and even then, check them off and write NA next to that step in the instructions.

Resist the urge to "add more lights" or "more channels." The urge will happen to you – it always does. More lights are just that – *more*, and *more* does not equate to *better.* A wonderfully creative and tasteful display can be done with only 8 or 16 channels and more often than not, the best displays are economical. The old *KISS* principle (Keep It Simple, Stupid!) is probably the single, best piece of advice a new DIY lighting enthusiast can follow.

Common Electronic Parts for many DIY Electronic Projects

- Resistors (1/4 watt): 120, 180, 680, 750, 1K, 27K ohm
- Resistors (1/8 watt): 120, 330, 470 ohm
- Resistors (1/2 watt): 8, 9, 10, 15k ohm
- Diodes: 1N4001, 1N4002, 1N4004
- Diodes – Zener type: 3.3 volt, 4.3volt, 9.1 volt, 5.1 volt (1 watt rating if possible)
- LEDs – red or green, 3mm type, for power/diagnostics use, not for displays
- Voltage Regulators: 3.3v, 5v, 12v, look for 1amp or greater
- Bridge Rectifier: DF01M, DF04M
- Non-polarized capacitors (ceramic or disc): .1uf, 1uf (50volt rating is usually ok)
- Polarized capacitors (electrolytic): 10uf, 47uf, 100uf, 2200uf, 4700uf (25volt rating is usually ok)
- DIP sockets: 6-pin, 8-pin, 14-pin
- Header pins (breakaway type, often available in rows of 40 or 80 pins)
- Female XBee header mounts (wireless users)
- Fuses, fuse holders: 5x20mm, 4amp, 5amp, 7amp
- RJ45 jacks (vertical and horizontal PCB mount type)
- 18.432mhz crystal oscillators
- IC Chips: H11AA1, PIC16F688, MOC3023 or MOC3023M, ST485BN
- Jumper shunts (Mouser # 151-8000-E)
- Heat sinks (for TO220 style voltage regulators – Mouser # 532-577102B00)
- Female header/wire housing sockets (make your own jumper wires – Mouser# 538-16-02-0102)

Good sources of electronic parts/supplies for DIY lighting projects (some suggestions are included):

- www.mouser.com (most comprehensive supply of all products)
- www.digikey.com (broad product line, good prices on resistors in bulk rolls)
- www.alliedelec.com (broad product line, similar to Mouser)
- www.futureelectronics.com (more electronic components, good prices)
- www.futurlec.com (preassembled adapter boards, some parts)
- www.taydaelectronics.com (factory extras/overstocks on parts, terminal blocks, fuses)
- www.newark.com (more electronic components, good prices)
- www.ebay.com (breakaway male header pins, DIP sockets, terminal blocks, LEDs, bulk capacitors)
- www.radioshack.com (excellent solder, transformers, hand tools, DVM's, wire)
- www.sparkfun.com (source for XBee programming tools, lots of adapters, multi-wire connection cables)
- www.seeedstudio.com (multi-wire connection cables, other adapters, retail adapters, cables)
- www.monoprice.com (cable adapters, bulk cat5 cable, connectors, cable adapters)
- www.sureelectronics.com (power supplies, adapters)
- www.jameco.com (parts, kits, crystal oscillators)
- www.wireframedlites.com (high-quality wire frame displays)
- www.creativedisplays.com (quality lighting accessories, wire, connectors, LEDs, ask for Paul Sessel)
- www.terryslightedchristmas.com (high-quality, custom made wire frame displays)
- www.wlcventures.com (FM radio transmitters, DIY lighting kits, controller enclosures)
- www.diyledexpress.com (LED controllers, FM radio transmitters, controller enclosures)
- www.holidaycoro.com (coroplast signage, other display items made out of corrugated plastic)
- www.n7xgshop.net/shop (Simple8, Simple16, Simple32 controllers, SSRs, more)
- www.diychristmas.org/store (DIGWDF wireless controllers, bare boards, prototypes, an awesome store!)