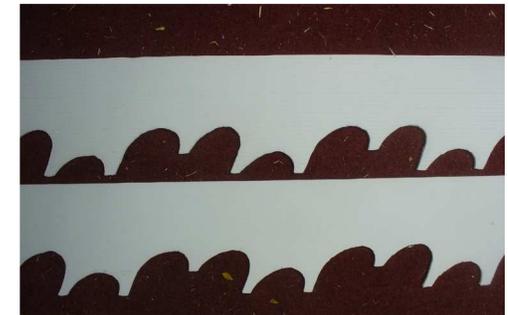
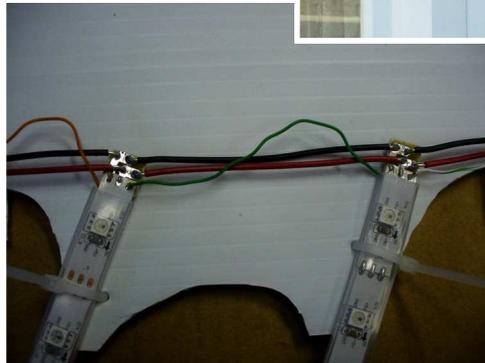
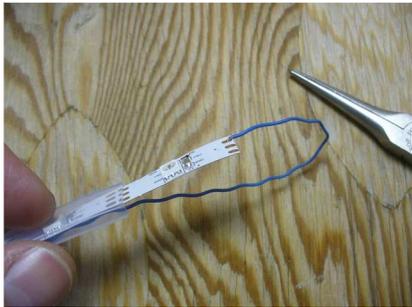
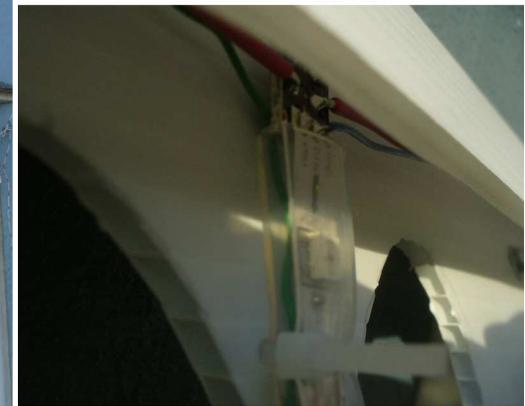


## How I Did – WS281X RGB Icicles



## Table of Contents

1. Intro.....	3
2. Overview.....	3
3. Design Considerations.....	5
3.1. Type of LEDs.....	5
3.2. Icicle Spacing.....	6
3.3. Power Distribution.....	6
3.4. Angles.....	7
3.5. Mounting/Appearance/Waterproofing.....	8
3.6. Diffusion.....	9
3.7. Surprises.....	10
4. Construction.....	10
4.1. Making the Icicles.....	10
4.2. Wiring the Icicles.....	14
4.3. Mounting.....	23
5. Usage.....	30
5.1. Installation.....	30
5.2. Power Supplies.....	34
5.3. Data Lines.....	35
5.4. Sequencing.....	36
5.5. Storage.....	37
6. More Info.....	38

Version 1.0 2/20/17

Copyright © 2017 eShepherdsOfLight.com

Permission granted to copy or distribute for non-commercial use.

## 1. Intro

The monochrome incandescent Chase-cicles I made many years ago<sup>1</sup> worked well, and since LEDs are becoming less expensive I thought I would try some RGB Icicles using WS281X LED strip. RGB Icicles allow more colorful effects, and individually addressable pixels allow text and graphics to be displayed, making the Icicles a versatile display prop.

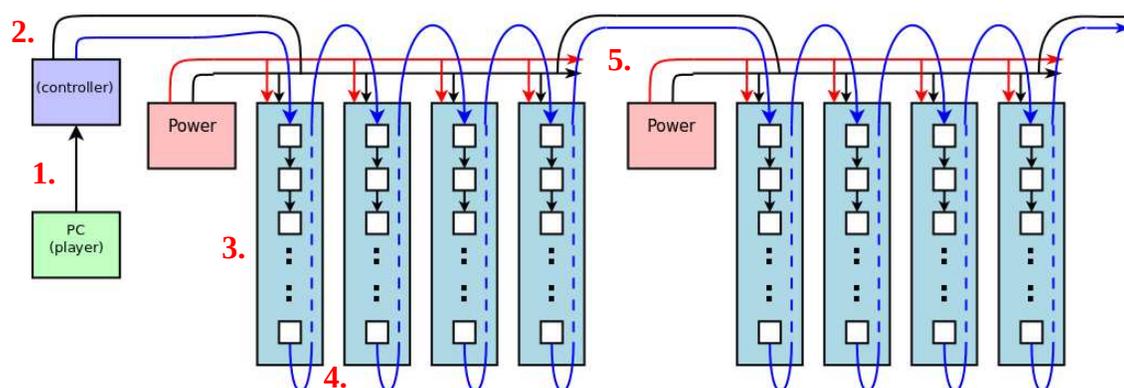
Building the Icicles took a fair amount of effort, but I was pleased with the results. I actually built the RGB Icicles 3½ years ago and am just now getting around to documenting them. ☺ This was one of my earliest WS281X projects, and I was unsure how durable they would be so I wanted to put some mileage on them before documenting my construction techniques. (That’s my story, and I’m sticking to it ☺) Now that I know they are durable<sup>2</sup>, I think it’s safe to write them up.

So finally, here is how I made some WS281X-based RGB Icicles. The discussion will be focused on the design and actual construction techniques and some tips and tricks I found along the way, not so much on the technical details of how WS281X LED strips actually work internally – that is already well documented out on the web. I actually have a mix of WS2811 / WS2812B. I treat them interchangeably and will refer to them as “WS281X”.

Please note that these techniques are not guaranteed to produce good results for everyone – they worked well for me in my environment, but YMMV.

## 2. Overview

First, let’s start with a high-level view of how the RGB Icicles are organized, and then drill down into the details in the remainder of this document.



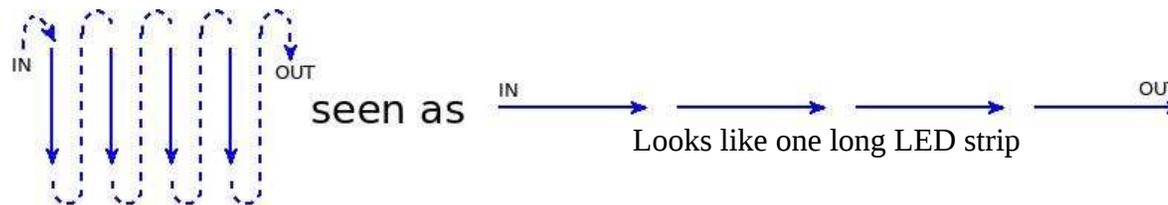
*Icicles - system overview*

1 Described at <http://downloads.eshepherdsflight.com/Howidid-ChaseCicles.pdf> (2008)

2 Of 2,070 pixels, I finally had 1 joint between a couple of the LED pixels fail in the 4th season of usage.

Data flows through the Icicles along the numbered path as follows:

1. A PC or laptop (or Raspberry Pi) runs some playback software to send sequencing data to a controller (if present<sup>3</sup>), in time to the music.
2. After receiving data from the PC, one or more controllers generate a WS281X data signal and send it to the RCB Icicles. If all the Icicles are daisy chained into one long string (as shown above with the **blue** line), then only one data signal is needed, otherwise multiple separate data streams would be needed. The controller and LED strip must have a shared ground (**black** line) to correctly interpret the data signal.
3. The WS281X LED strip acts like a very long *serial shift register*. It is cut into short pieces to form Icicles, with data traveling “down” each Icicle from one pixel to the next. A single wire carries combined data and clock signals, so it has somewhat strict timing requirements although the protocol itself is fairly simple<sup>4</sup>.
4. At the end of each Icicle, the data out loops back up to the start of the next Icicle. The controller is unaware of this and sees the daisy chained Icicles as one long WS281X LED strip. However, in the sequencing software the Icicles would be modeled as a large left-to-right, top-to-bottom grid:



5. Power usage adds up when a lot of LEDs are on at the same time, so the Icicles are grouped according to the power supply’s capacity. Separate power bus lines feed each group of Icicles (**red** and **black** lines). The data line can be daisy chained to the next group if desired, along with a common ground reference.

<sup>3</sup> This past season I skipped the discrete controller and just used the the RPi video core (GPU) to generate the WS281X data signals.

<sup>4</sup> For protocol and timing specs, see <https://cdn-shop.adafruit.com/datasheets/WS2811.pdf>

## 3. Design Considerations

There are a number of design choices that must be made up front. Below are the ones I used.

### 3.1. Type of LEDs

Sometimes the choice of LEDs is dictated by design constraints, but otherwise it's just personal preference. I chose 5m rolls of 5V WS281X LED strip with 30 LEDs/m<sup>5</sup> ( $\approx 16\frac{1}{2}'$ , 1.3" spacing) for several reasons:

- For displaying text and simple graphics on the Icicles, it seemed like close, uniform horizontal and vertical spacing was more suitable. This made LED strip preferable over LED strings.
- My initial experiments with WS281X pixels showed them to be reliable and easy to control<sup>6</sup>, so I have standardized on those over the years. I have a mix of WS2811 and WS2812B LED strip in the Icicles, and they seem to be somewhat interchangeable.
- At the time, the 5V version of WS281X LED strip offered individually addressable LEDs while the 12V version did not (3 LEDs per chip). I wanted individual control of the LEDs so I went with the 5V strips.
- I had some old PC power supplies, and those appeared to be able to pump out higher current at 5V than at 12V, so I would need fewer power supplies. (some 12V LED strip is less power efficient than 5V).
- The power supplies could be mounted physically close to the Icicles, so voltage drop losses due to the resistance in long power leads was not an issue for me.

BTW, I had originally intended to build the Icicles using my own custom-designed chipplexed LED strips using SMD 5050 LEDs (same kind of LEDs found in WS2811 LED strip), but the price of pre-made WS281X LED strip came low enough that the extra effort wasn't really worth it.



*One segment of custom chipplexed SMD5050 LED strip*

<sup>5</sup> <https://www.aliexpress.com/item/5m-WS2811-LED-digital-strip-60leds-m-with-30pcs-WS2811-built-in-the-5050-smd-rgb/653641294.html>

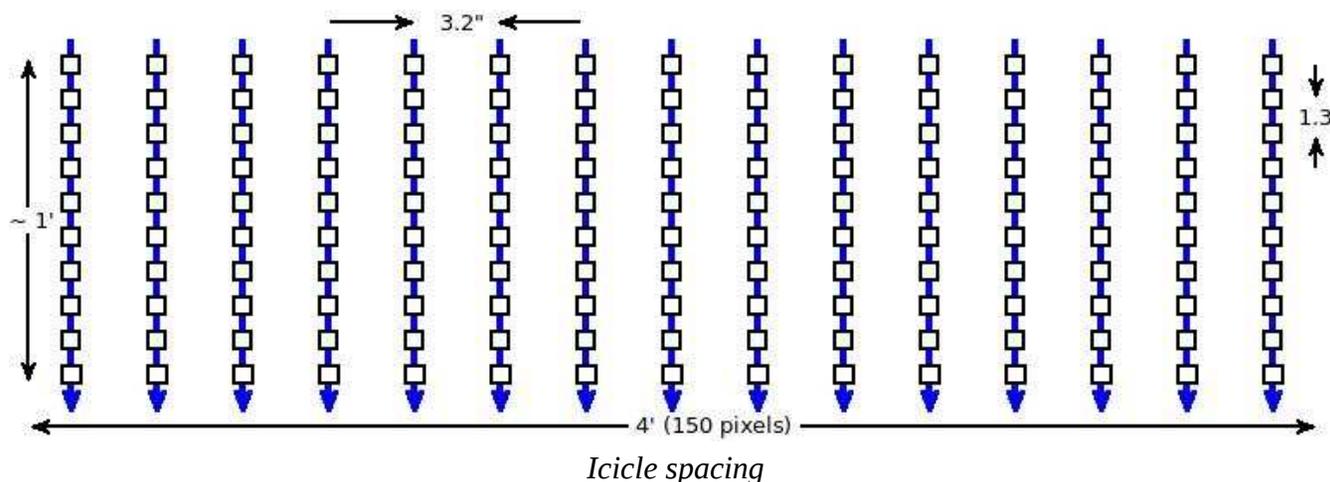
<sup>6</sup> I was able to control them reliably using the 4.6 MPS PIC16F688s in my DIYC Renard controllers.

## 3.2. Icicle Spacing

My goal was to display text and simple graphics on the Icicles, so I chose to make all the Icicles the same height. This allows them to be modeled in the sequencing software as a simple rectangular grid. The uniform height doesn't look as much like real icicles, but they could be lit up to appear unevenly using software.

Populating an entire roof line with Icicles would take a lot of LEDs, so I tried to keep the Icicles relatively short. A height of 10 pixels ( $\approx 1'$  tall) seemed like a reasonable compromise – that allowed a fairly large font and some icon-like graphics to be displayed, and was easily readable from the street. With 10 pixels per Icicle, a roll of LED strip would give exactly 15 Icicles.

I think Icicles look better with more space between them, but if they are too far apart the text and graphics are not as easy to recognize. After some experimenting, I chose 3.2" horizontal spacing which gives an aspect ratio of  $\approx 2.5:1$ . It also takes exactly 15 Icicles to make each 4' section of roof line, making layout calculations a little easier. (Coroplast also comes in 8' sheets, which was another consideration).



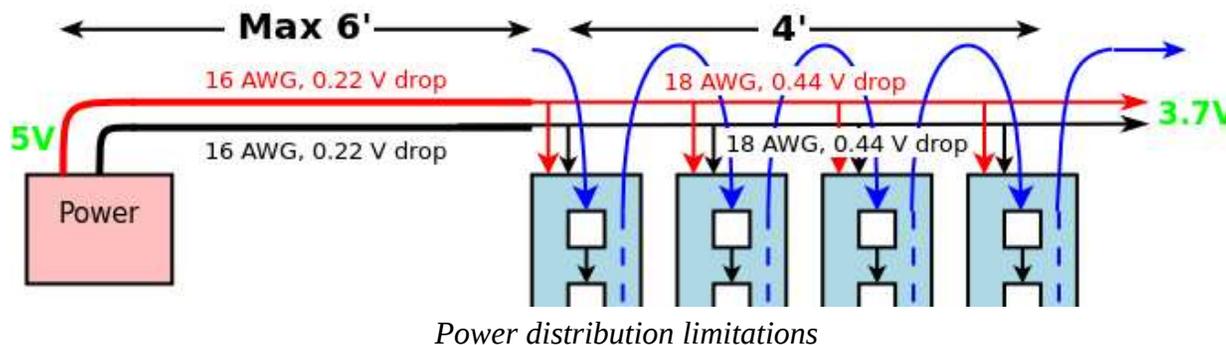
## 3.3. Power Distribution

Each 4' section of Icicles contains 150 WS281X pixels. At full white, current drawn is  $150 * 60 \text{ mA} = 9\text{A}$ , but all white is a rather boring pattern so I don't actually ever use that. In fact, I now limit white to  $\approx 83\%$  intensity in software because it's just too bright when that many pixels are packed close together. This also gives a little extra safety margin, so a 20A power supply can safely power 8' of Icicles (300 pixels @83% white  $\approx 15\text{A}$ ).

Based on the resistance and current carrying capacity for various wire gauges<sup>7</sup>, I decided that the power supplies could be up to 6' away from the Icicles using 16 AWG, and the internal Icicle power bus could be 18 AWG:

- power supply to first Icicle using 16 AWG: 6' x 4 mΩ/ft = 24 mΩ resistance
- power bus within each 4' section using 18 AWG: 4' x 21 mΩ/ft = 84 mΩ resistance
- total voltage drop: 9A x 2 x (24 + 84) mΩ ≈ 1.3V max voltage drop (round trip)
- 5V - 1.3V = 3.7V = just enough for full white<sup>8</sup>

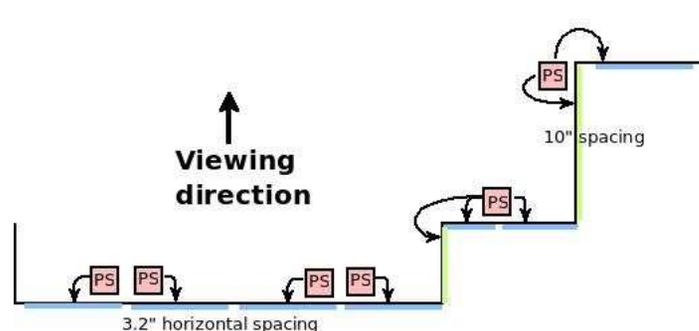
So the overall power distribution worked out as follows:



In this arrangement, the worst case voltage to the furthest WS281X nodes is 3.7V which is just barely enough. The wire gauge and length can be adjusted as needed to stay within the target voltage drop limits.

### 3.4. Angles

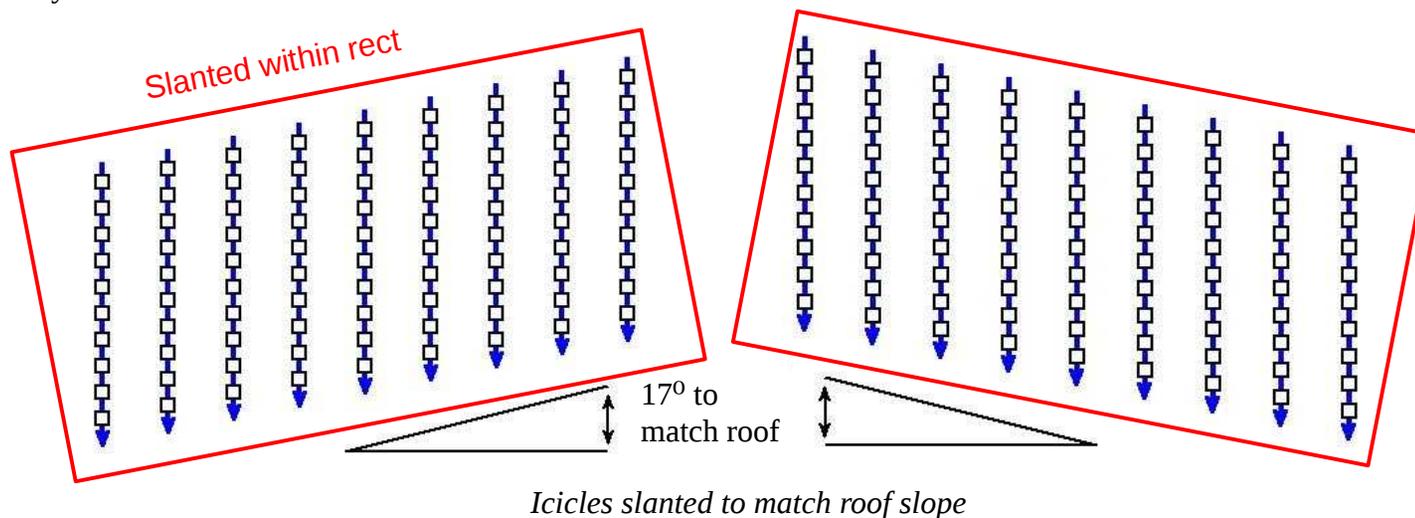
Most of my front roof line is parallel to the street so I used the 3.2" horizontal Icicle spacing described earlier (blue sections in the diagram to the right). However, some areas are perpendicular to the street, so I triple-spaced the Icicles in those areas to reduce pixel count (green sections, right):



<sup>7</sup> There is a nice wire chart at [http://www.powerstream.com/Wire\\_Size.htm](http://www.powerstream.com/Wire_Size.htm) (I used the “chassis” numbers)

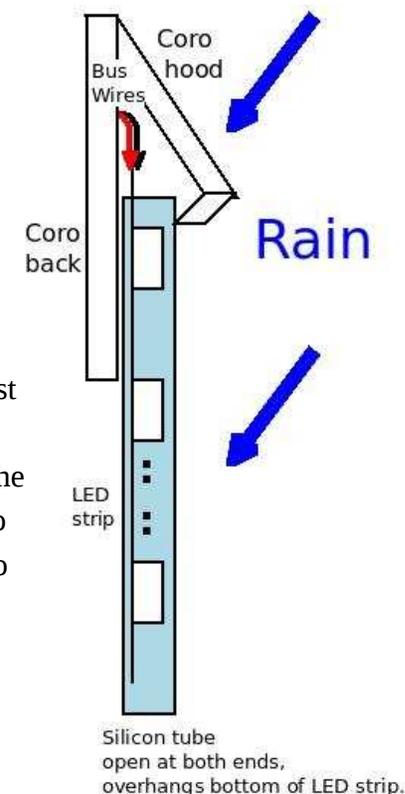
<sup>8</sup> If the voltage of WS281X pixels is too low, they look pinkish as the green and blue fade (require more voltage than red).

My roof also slopes mostly in the same direction as the street, so I had to slant many sections of Icicles to match the slope of the roof (so they would appear to be hanging straight down rather than leaning). This was easy enough to do, but it does mean that the Icicle sections are not interchangeable – they must be used on a section of roof line that matches their slant.



### 3.5. Mounting/Appearance/Waterproofing

I wanted a rigid, light-weight material to permanently fasten the Icicles to, which could be put up and taken down easily. There were several materials that seemed suitable: wood, PVC, coroplast, etc. I decided to use white coroplast (corrugated plastic) because it is light-weight, waterproof, reasonably sturdy (if not too much weight on it), not too expensive, readily available, and easy to cut. The ease of cutting was an important consideration because I wanted the Icicles to look somewhat like they were dripping out from a layer of snow on the roof above them. The Icicles had to be permanently mounted on a rigid structure in order to prevent damage and to make setup quick and easy. They also needed to be waterproof because we get rain (unless there's a drought ☺). However, I also wanted a snowy appearance around the tops of the Icicles. The obvious choice was to use white coroplast.





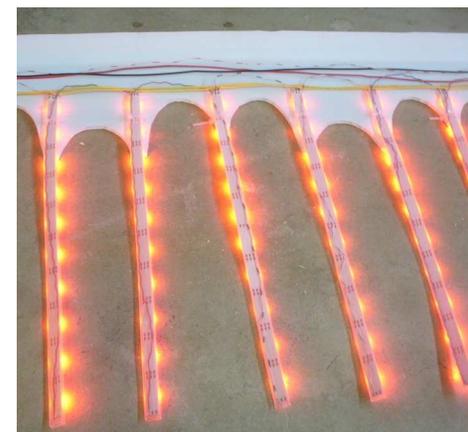
*White coroplast for snow/ice appearance, structure, and waterproofing*

My waterproofing strategy was to just use gravity and the silicon tubing itself to keep rain off the LED strip. I read some forum posts that suggested that you don't want moisture trapped inside the sealed silicon tube, so I actually just left the tubing **completely open** on both ends to allow air flow, and then I just ensured that water could not get inside the tubing by putting a little coroplast hood over it.

### 3.6. Diffusion

WS281X can be quite bright to look directly at, so I experimented with a layer of coroplast in front for diffusion and even turning them backwards<sup>9</sup>.

In the end, I decided to just point them forward and dim them if needed:



<sup>9</sup> This worked very well on the garage door. See page 10 of <http://downloads.eshepherdsflight.com/Howidid-DumbRGBPixelGrid.pdf>

## 3.7. Surprises

I thought I had carefully planned this Icicle project and taken everything into account. Testing went well but I still ran into a couple of surprises later during final show testing:

- My garage door remote would not work when the Icicles were running. It turns out that WS281X pixels generate some RF noise while the LEDs are on (or maybe only during data transfer). I had a nice row of Icicles along the roof line of the garage, which was the perfect place to cause line-of-sight interference with the garage door remote. ☹️ So, I had to go back to the old days and actually get out of the car to open the garage door when the LEDs were on. 😊
- My FM transmitter also would not work. I had foolishly connected the FM transmitter to the same power supply as some of the WS281X Icicles. Bad move – the FM now carried a hum rather than the music. ☹️ This actually turned out to be useful for a little while – I could tell by the tone of the hum on the FM radio which pattern was displayed on the Icicles without even looking at them. 😊

## 4. Construction

The construction techniques I used are described below. This might not be the best way to do it, but it worked well for me. There were 2 main phases of construction: preparing the segments of WS281X LED strip, and then mounting and connecting them.

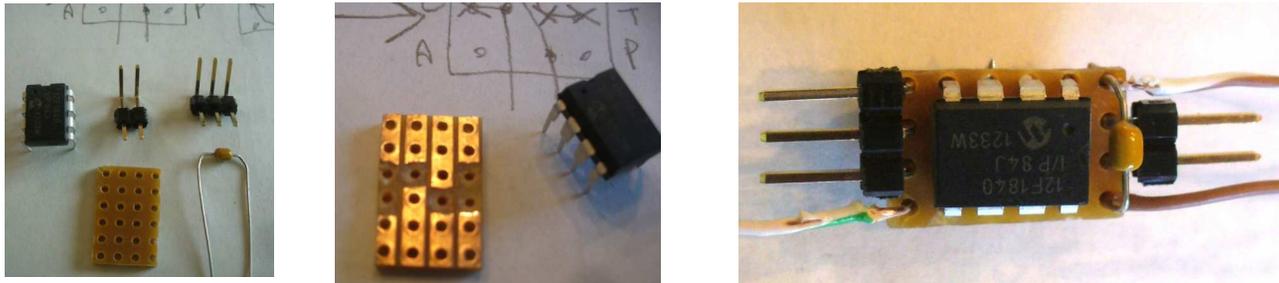
### 4.1. Making the Icicles

Cutting/prepping/wiring the LED strip went fairly quickly after I got the hang of it.

Tools and materials I used for this step:

- one 16.5' roll of WS281X LED strip for each 15 Icicles (if using 10 pixels per Icicle)
- 1½' of wire for each Icicle (I used one strand of 24 AWG from a cat3 or cat5 cable)
- Exacto knife (to cut the outer silicon tube without cutting the LED strip)
- scissors (for cutting the LED strip)
- soldering iron, solder, maybe also de-solder wick
- wire cutters, strippers
- push pin or needle (to enlarge the LED strip holes slightly)
- WS281X controller and test leads for testing + test leads
- 10A or better power supply (for testing)
- weights or light clamps can be useful (inside-out clothespins work well)

I first tested the rolls of WS281X before making any cuts, so I could return any that didn't work. Any controller compatible with WS281X can be used. I made a little PIC12F1840 circuit for testing so I didn't need a PC on the work bench (I was originally going to make a bunch of these to drive the Icicles, but changed my mind):



*\$1.50 USD WS281X controller/tester*

After verifying that the LED strip worked, it was time to start cutting. I was a little nervous the first time, but I had extras in case I made a mistake. It turns out that WS281X LED strip is actually quite easy to work with.

I cut the power leads off the first roll and then cut it into segments of 10 pixels each. Unfortunately, there was a solder joint every 15 pixels, so those joints required some extra effort to desolder and clean up so they would accept wires and solder:



*LED strip solder joint interfered with cut points*

After dealing with those solder joints a few times, I started cutting only one pixel off at the beginning of each LED strip, and then every 10 pixels thereafter. This way I never ran into a solder joint at the place where I wanted to cut. 😊 The last piece, of course, only had 9 pixels so I soldered that first cut-off pixel onto it to make the correct length Icicle. To ensure the silicon tube was long enough, I had to remove the silicon end cap (saved it for other use), dig through the silicon glue, then pull the LED strip out of the tube far enough to cut off the first pixel:



*Cut off first pixel to avoid solder joints on later cuts*

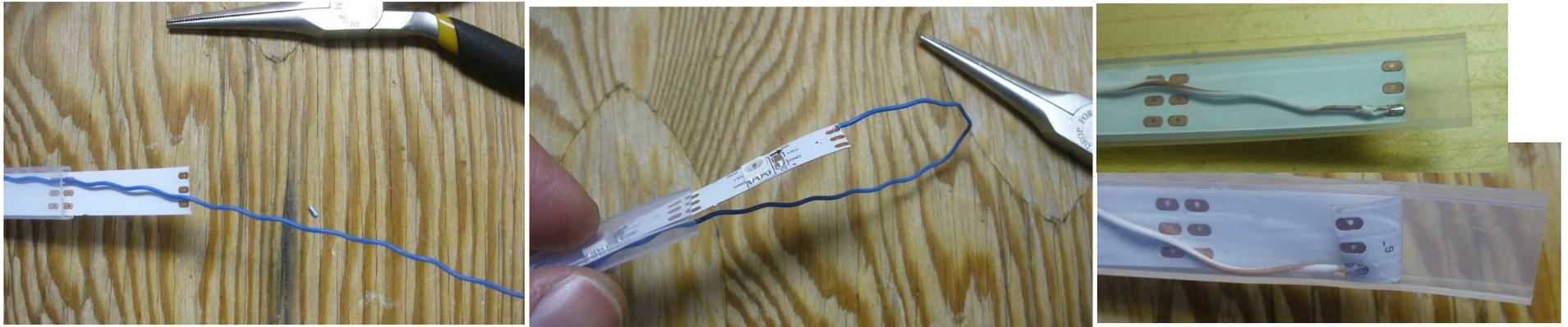
Some rolls had more silicon tubing on the trailing end than others so sometimes I did not need to pull the LED strip out before cutting off the first pixel. For other rolls, I only needed to pull it a little. In all cases, I wanted to end up with the silicon tube covering the full length of Icicle. When reattaching the first pixel, I used short pieces of AWG 24 wire then soldered over them:



*Solder first pixel back onto last piece of LED strip to make a full-length Icicle*

After the LED strip was cut into the correct lengths, I then added a data return wire to allow the Icicles to be daisy chained. 24 AWG seems to be about the right size wire, although I did need to enlarge the holes in the LED strip using a pin.

I threaded a strand of cat5 behind the LED strip (actually, cat3 is easier since it is not as twisty) and hooked it onto the data output of the last pixel, pulled the LED strip out a little to allow soldering, then soldered. The first time I soldered it on the wrong side. d'oh. Rather than redoing it, I just folded the LED strip over at the end and then tucked it back into the tube:



*Thread a strand of cat3 or cat5 behind LED strip and solder to data out*

After all the data return lines were soldered, the Icicles were ready to be connected together and mounted. Note that the silicon tube was exactly the same length as the LED strip. When I pulled the LED strip up a little to allow for soldering at the top of the Icicles, this left a little extra silicon tubing at the bottom of each Icicle. This was by design, so that the LED strip would not be exposed to rain:



*Icicles cut and prepped; bottom end recessed within silicon tube*

## 4.2. Wiring the Icicles

Interconnecting the Icicles and adding power feeds was the most time-consuming part of the project. Below are the various techniques I tried. In most cases I worked on one 4' section of Icicles at a time (one roll of WS281X LED strip). A few times I had other quantities due to variations in the roof line.

Tools and materials I used were:

- Icicle segments from previous step
- 2 5' lengths of 18 AWG wire for power bus
- wire cutters, scissors, strippers
- soldering iron, solder
- weights or light clamps can be useful (inside-out clothespins or Third Hand)

### 4.2.1. First Technique

My first technique, described here, was to connect a pair of power bus wires directly to the LED strip. This was rather tedious and probably stressed the ends of the LED strip more than necessary, but the Icicles have worked reliably for 4 seasons so I guess it wasn't too bad.

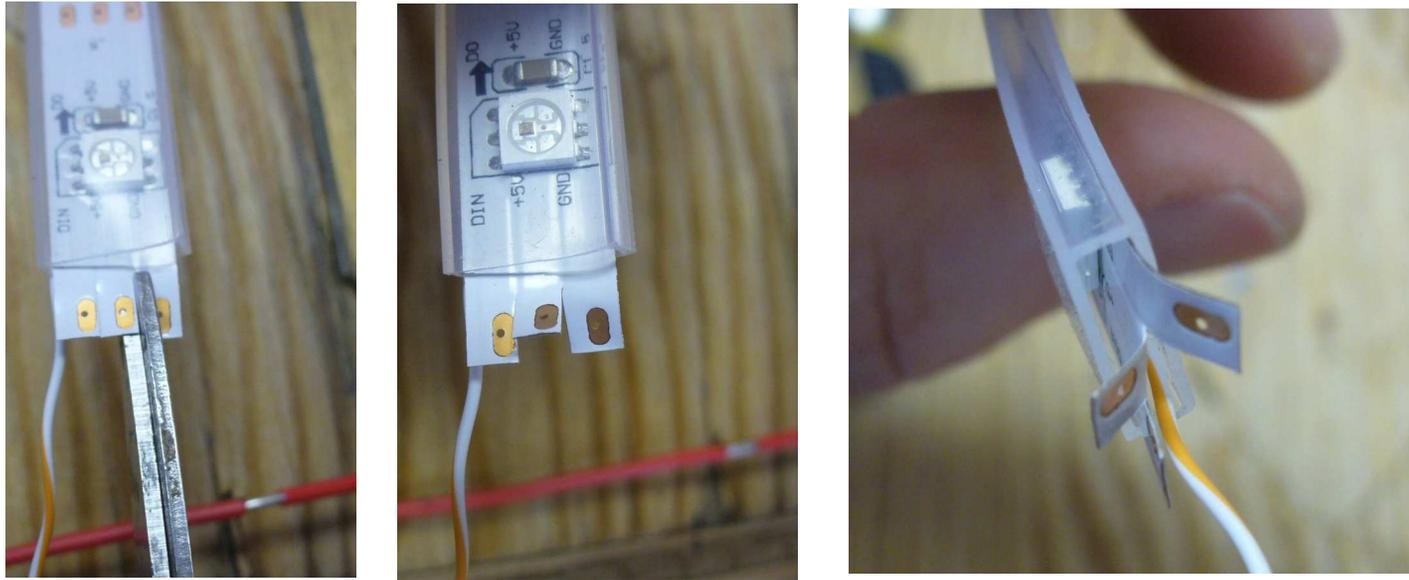
I cut a pair of 5' power bus wires from rolls of AWG 18 solid wire, one red and one black (to distinguish polarity). I used solid core wire to help keep the connected Icicles in a rigid formation. Only 4' was actually needed between the Icicles, leaving 1' extra for connecting to the power supply leads.

I stripped 2" of insulation from one end of the bus wires, which allows  $\frac{1}{8}$ " for each Icicle ( $15 \times \frac{1}{8}" \approx 2"$ ). Then I marked the remaining insulation at  $\frac{3}{16}$ " intervals and used wire strippers to cut and slide the insulation apart  $\frac{1}{8}$ " at each of those places ( $3 \frac{1}{16}"$  insulated +  $\frac{1}{8}"$  gap  $\approx$  the 3.2" spacing needed for 15 Icicles to span 4')



*Using wire strippers to make  $\frac{1}{8}$ " gaps in insulation on the power bus wires*

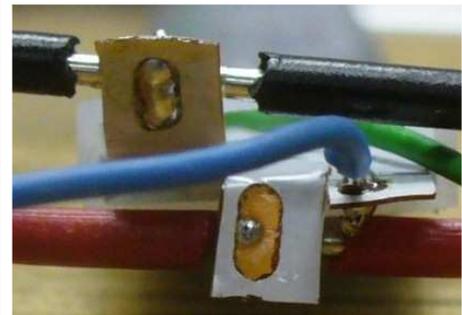
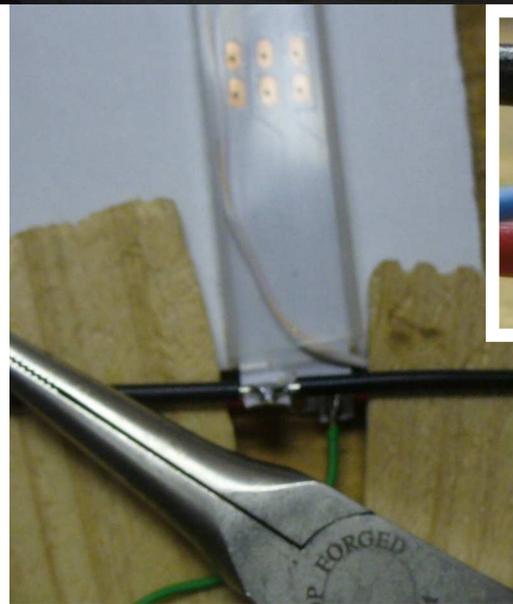
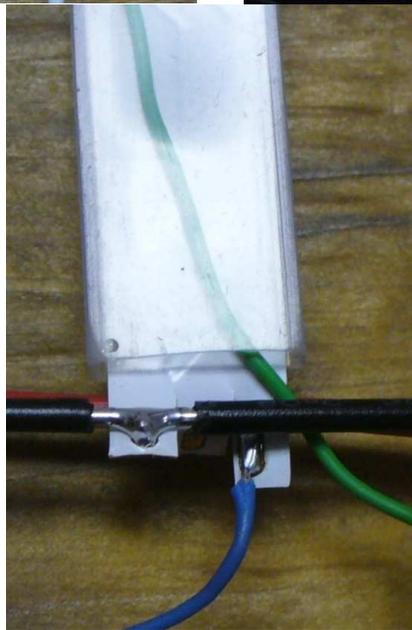
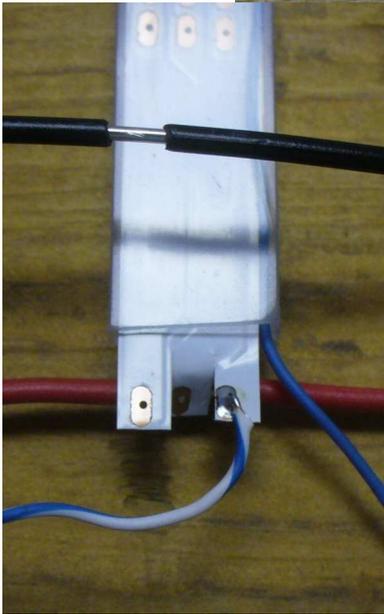
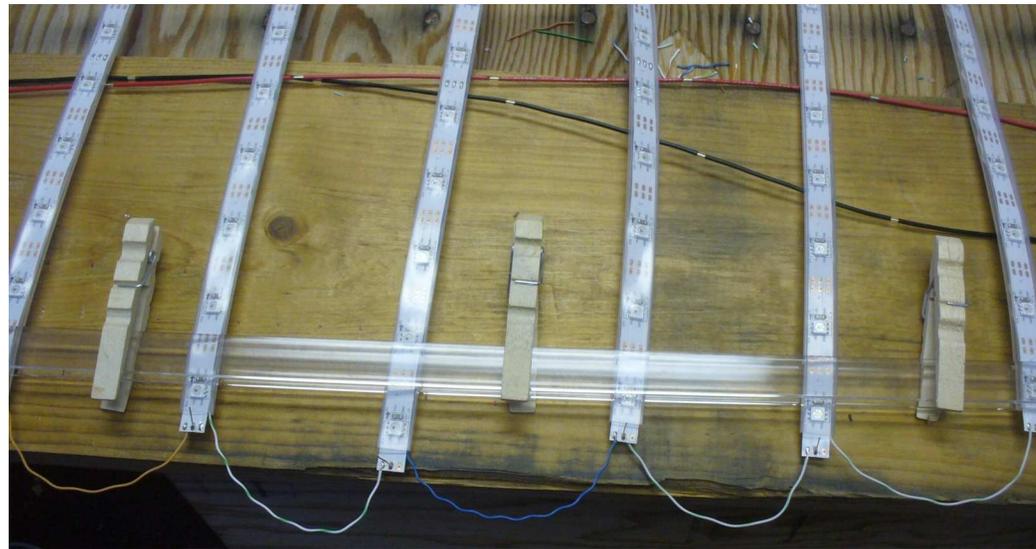
Next I pulled the LED strip out of the silicon tube to expose the top edge by  $\approx \frac{1}{2}$ ". This left the other end of the LED strip concealed within the silicon tubing to protect it from rain (explained under Mounting). Then I used scissors to make short cuts into the end of the LED strip and partially folded it apart to form solder tabs for the power bus wires:



*Solder tabs created for the power bus wires*

Next the bus wires needed to be soldered to the LED strips. The middle tab is the hardest to solder if the tabs on either side are already soldered, so I started with it. This turned out to be the +5V pad for the LED strip I was using - sometimes the connections are in different locations so check carefully before soldering.

I aligned the first gap in the bus wire with the middle tab and soldered it, and then went down the bus wire and did all the other Icicles as well. Then I clamped the Icicles together so the bus joints wouldn't move and flipped the whole assembly over. This avoided strain on the first bus joints and allowed me to more easily align the other bus wire with the ground pads and solder them (I also soldered the data return line from the previous Icicle onto the remaining solder tab). I used some shims to hold the second bus wire in place while soldering:



*Attaching bus wires, clamp and flip*

After the bus wires were all attached, I twist-tied them to some plastic slats to help keep the bus wires rigid until they were mounted to coroplast. The Icicles hung by the bus wires to prevent damage:



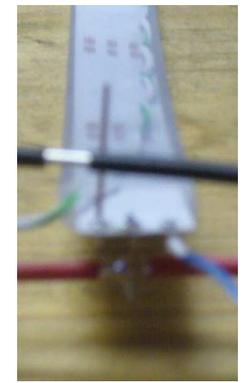
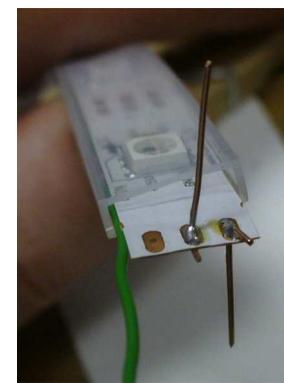
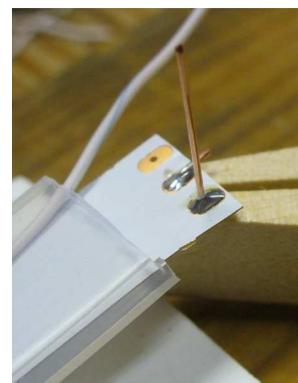
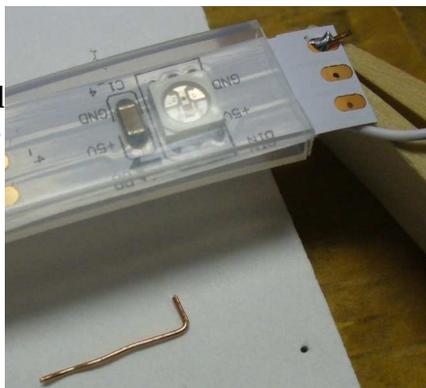
*Icicle assemblies twist tied to retaining rods and hung until mounted*

#### 4.2.2. Second Technique

That first technique worked but was rather tedious, and the solder tabs on the ends of the LED strip seemed too fragile (narrow) so then I tried a variation, which also worked but was not much easier.

Instead of cutting solder tabs into the ends of the LED strip, I soldered little stand-offs (short pieces of bare wire from a cat5 cable) to the LED strip, which allowed the bus wires to be separated and soldered without bending the LED strip:

I clamped, flipped, and hung the assemblies as described earlier.



*Using stand-offs instead of cutting solder tabs*

### 4.2.3. Third Technique

Next I tried to streamline the process and avoid the clamp and flip steps.

Rather than a pair of long bus feed wires, I cut individual pairs of bus wires (18 AWG) about 4" long and soldered them directly to the LED strip pads:



*Bus wires soldered directly to LED strips and then hung*

I then bent the bus wires at right angles near the end of the LED strip so the Icicles would be parallel to each other, temporarily twist tied the assemblies to plastic rod and hung them prior to mounting as with previous techniques.

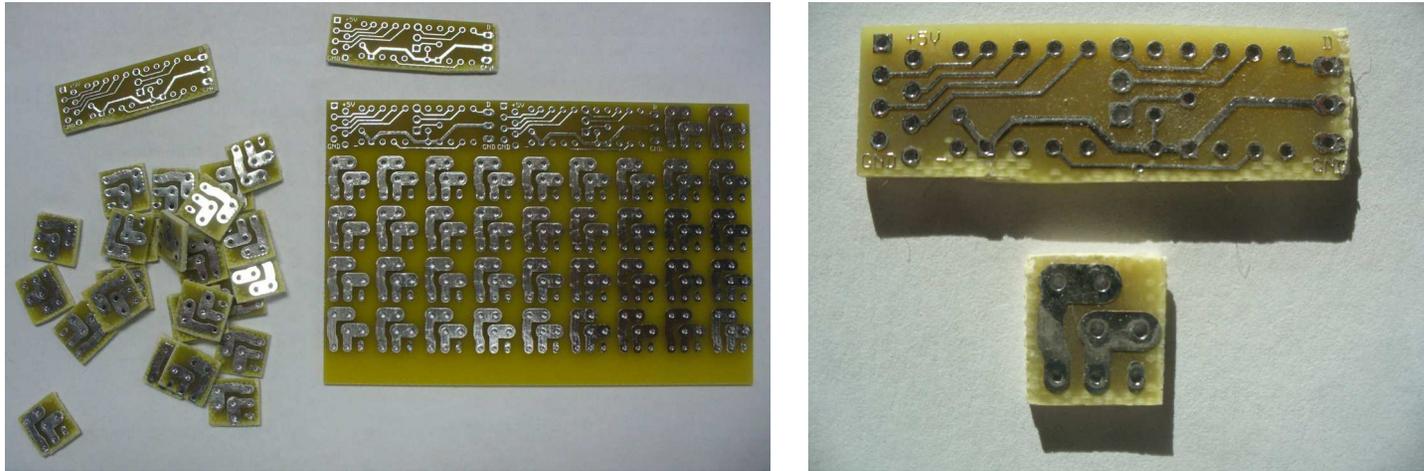
This technique was a little easier, but it took trial and error to get the bends just right, and I had to use 2 pairs of needle nose pliers (didn't want to strain the solder connection to the LED strip). Also, the 18 AWG did not fit into the tiny holes in the LED strip pads, so they were only soldered onto one side of the LED strip. (I preferred to have the power lines soldered to both sides of the LED strip to ensure better conductivity).

### 4.2.4. Fourth Technique

I then decided to try a little PCB magic to cut down on the awkward steps. I used ExpressPCB<sup>10</sup> to make some right-angle power bus connectors that could be soldered to the end of the LED strips, and I also included a version of my PexRenard<sup>11</sup> inline mini controller board (at the time, I intended to use a PIC12F1840 chip to drive each group of Icicles):

<sup>10</sup> Software and mini-board service from <https://www.expresspcb.com/miniboard-standard/>

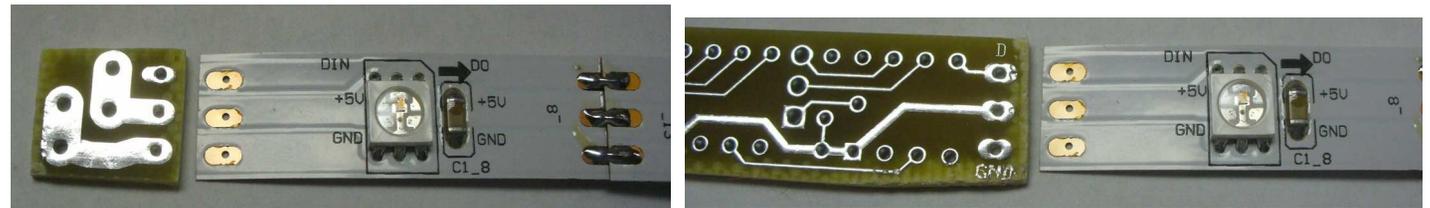
<sup>11</sup> A very narrow Renard-compatible controller that fits inside 1/2" dia Pex tubing.



*LED strip connectors and PIC12F1840 controller mini PCBs*

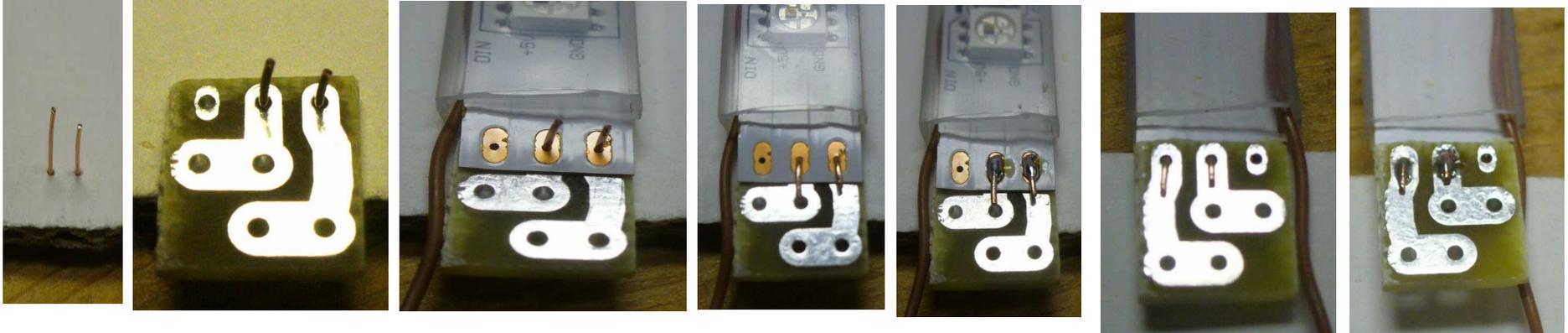
For prototyping, I usually skip the solder mask and silk screen to cut down on costs. I drew up the PCBs with ExpressPCB and had some prototypes made. Each mini proto board was big enough to hold 38 LED strip connectors + 2 mini-controller boards. The ExpressPCB mini proto service gives 3 boards, so I had enough to make 114 Icicles. I used some tin snips to cut the PCBs apart – FR4 is tough to cut.

Using the connector PCBs made it quite a bit easier to attach power bus wires to the Icicles. I designed the pads on the connector and the controller to line up with the pads on the LED strip:



*PCB-ette pads line up with LED strip pads*

Attaching the power connectors to the LED strip went fairly quickly. I jabbed a couple short pieces of bare cat5 into a piece of cardboard, dropped on the connector PCB and then the LED strip, bent over the wires, and then soldered. I then flipped it over, bent the wires over and soldered the other side (I prefer to solder both sides of the power pads on LED strip for safety). I had a whole pile of them done pretty quickly:



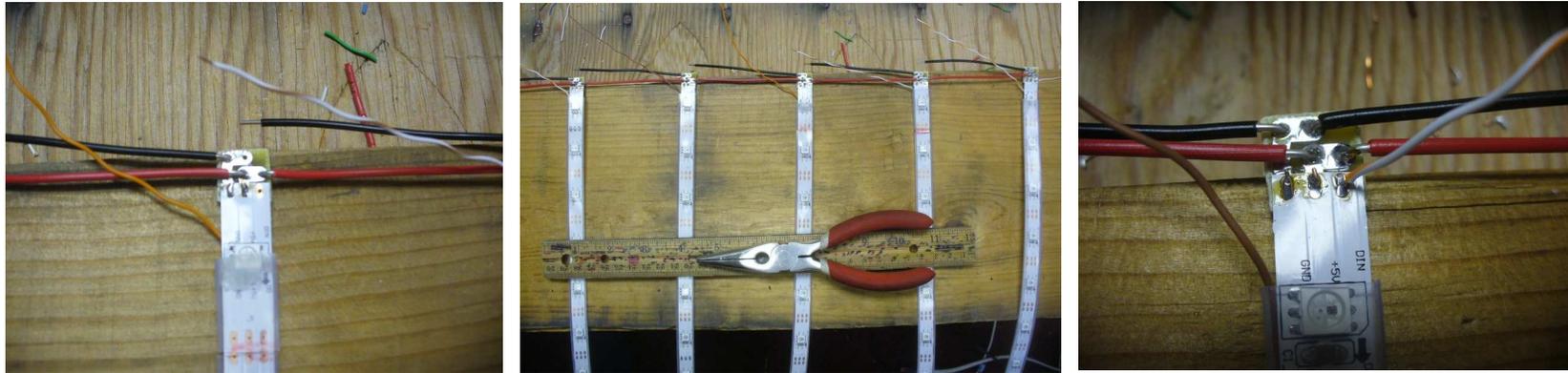
*Soldering power connector to end of Icicles*

Then I cut 18 AWG red and black bus feeders (as with the previous technique), bent over the ends, and attached a pair to each Icicle. I tried a couple of different ways, but I found always going the same direction to be the easiest. Either way, I found that I could solder up a bunch of them at the same time:



*Attaching bus feeder wires*

Then I started connecting the individual Icicles together, building up the assembly one Icicle at a time. It was fairly easy to keep the Icicles aligned correctly by doing one lead, then adjusting before soldering the second:



*Soldering one power lead at a time, then adjust before soldering second lead*

I found that these power PCB-ettes actually strengthened the end of the LED strip. I no longer needed to twist tie them to a plastic rod, and could handle or temporarily hang them just by the power bus, which simplified things further.

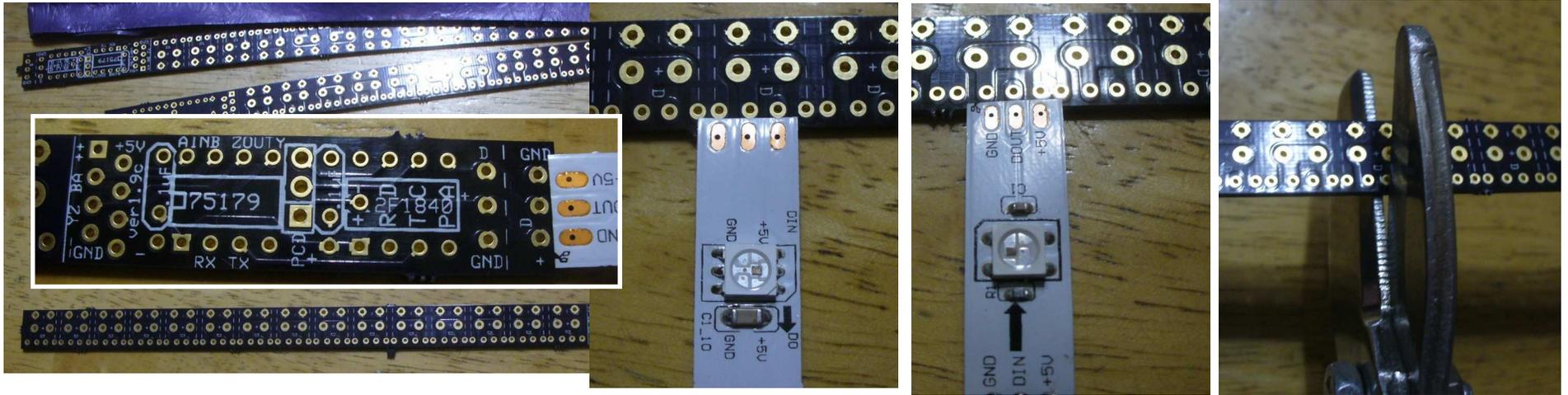
#### 4.2.5. Going Pro!

The mini PCB connectors made construction quicker and easier, so I decided to “go pro” and have a bunch more fabricated. This time I reorganized the mini PCBs into a “package” that had 15 connectors and one mini controller – enough for one full 4’ section of Icicles.

By this time I had a mixture of WS2811 and WS2812B LED strip. They appeared to have similar operating characteristics, but I discovered that the pads were in a different order! I wanted only one version of PCBs, so I modified the PCB-ette connectors to accommodate either pin-out. This turned out to be an interesting little puzzle but it had a simple solution: I reworked the PCB-ettes into one long continuous PCB and then added cut marks so the PCB-ettes could be cut to fit either pin-out. On the mini-controller, I added a second set of pads so that the end of the LED strip could be connected to either pinout. I used ExpressPCB again, but I generated Gerber files<sup>12</sup> this time and send them to OshPark<sup>13</sup>:

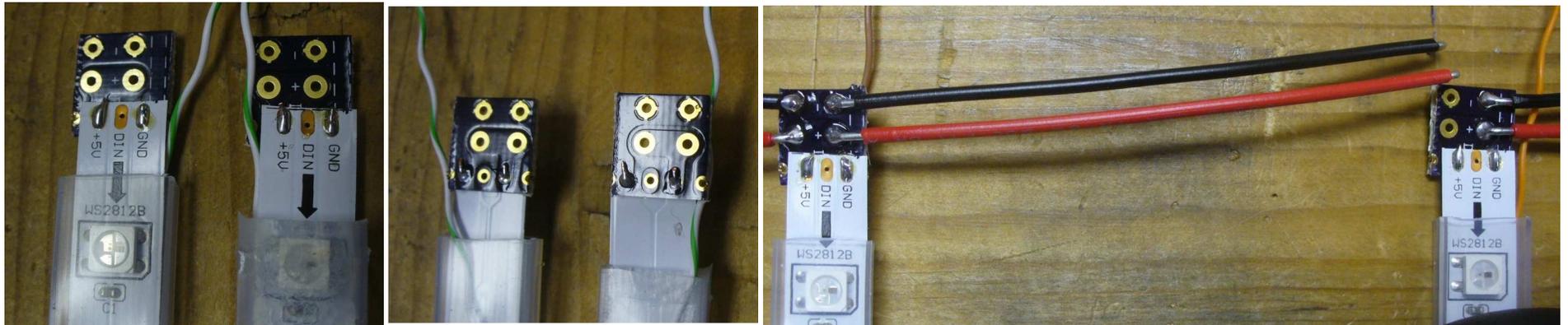
<sup>12</sup> I used the PDF to Gerber converter script from <https://github.com/swannman/pdf2gerb>

<sup>13</sup> Purple PCBs look cool, too. <https://oshpark.com/>



*PCB-ettes accommodate either LED strip pad order; choose cut lines as appropriate*

So now I could make Icicles using WS2811 or WS2812B LED strip fairly quickly and easily using only one version of PCB-ettes. Construction of the Icicle assemblies proceeded as before, except faster now: 😊



*PCB-ettes attached to LED strip*

If anyone would like to use the PCB-ettes (for non-commercial purposes), the Gerbers are at:

[http://downloads.eshepherdsoflight.com/PexRenMini+Strip\(ver1\\_1\).zip](http://downloads.eshepherdsoflight.com/PexRenMini+Strip(ver1_1).zip)

I suppose we could fabricate a bunch of these and do some kind of group buy if anyone is interested.

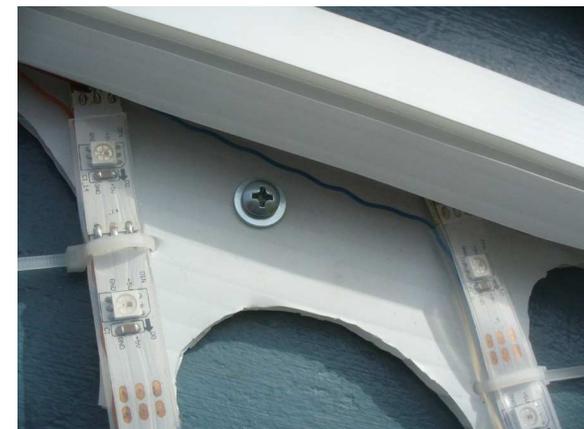
### 4.3. Mounting

After the Icicle assemblies were built, I fastened them to white coro backing so make them sturdier, waterproof, and look a little more like snow/ice.

Tools and materials used during this step:

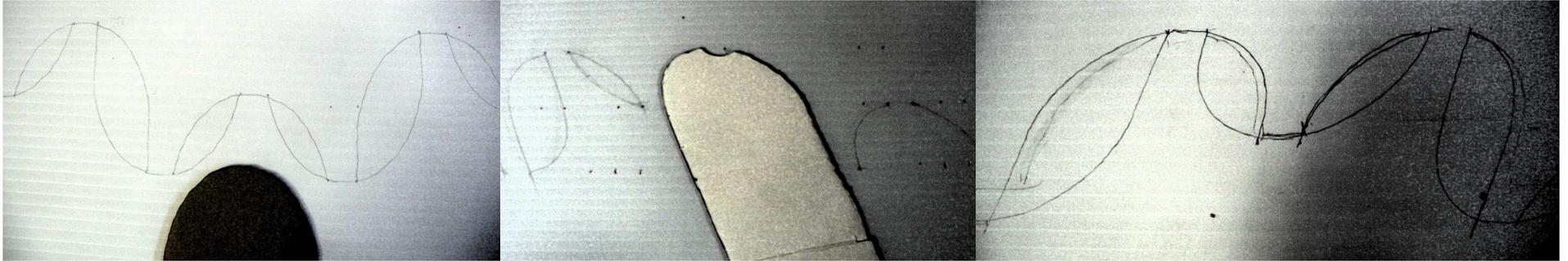
- 4' x 8' sheet of 4 mm white coro (~ 1½' x 4' used for each set of 15 Icicles)
- hobby knife
- awl or nail
- 4" zip ties (clear/white to blend with Icicle)
- 4' of 2 conductor 16 AWG outdoor wire
- wire cutters, wire strippers
- soldering iron, solder
- some kind of connectors

I went with a “unibody” design with white coro providing both the mounting structure and the visual covering for the Icicles. This simplified construction and avoided the need for using an adhesive:



*White coro provides mounting surface and visual covering*

In order to get a uniform look, first I made a pattern. You probably don't need this if you are artistic or want a less uniform look. I tried to make it look like snow and ice built up between the Icicles so my template used a rounded end. I used 2 different versions: an upright pattern and a slanted pattern. I traced around these onto the coroplast with a pencil and then just moved it along, varying it up and down, to replicate it:



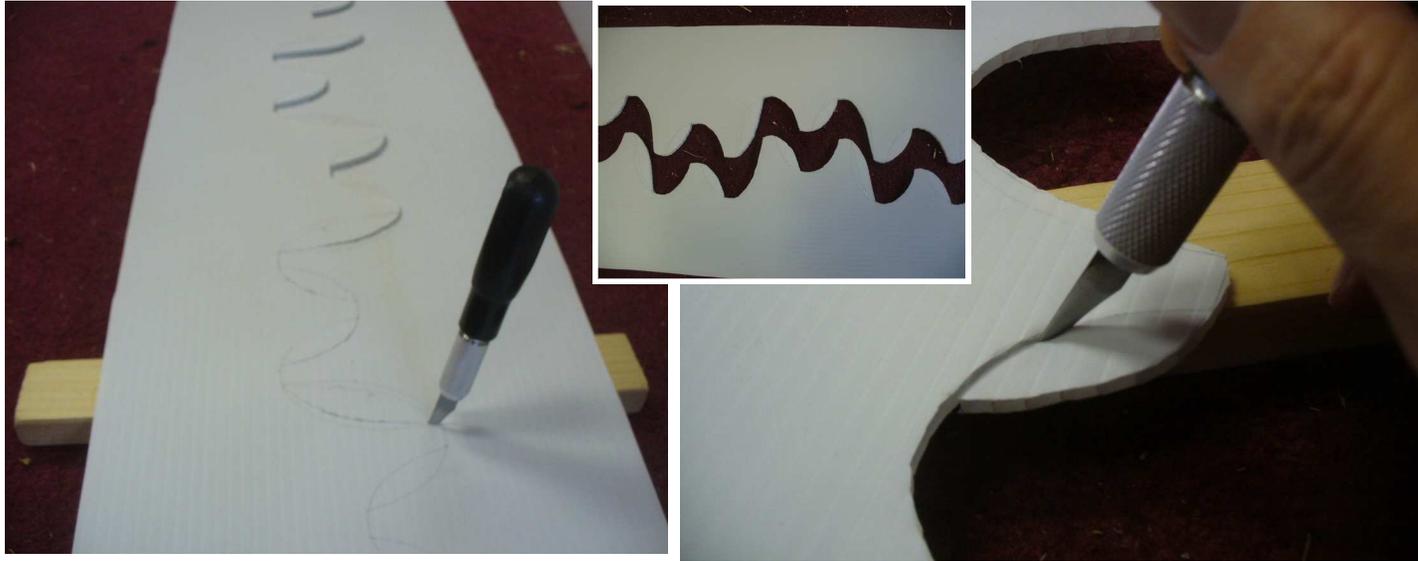
*Cardboard templates: upright and slanted versions*

Coroplast is not that expensive, but I tried to minimize wastage by interleaving 2 copies of the pattern - one upright and the other upside down. This gave 2 identical pieces, but used about 30% less coroplast:



*Interleaved 2 copies to reduce wastage*

Cutting the coro was a little tedious, but not too bad. Having a sharp hobby knife made it easier. After tracing the entire length of coro, I first cut apart the 2 pieces, then trimmed out the curved areas:



*Cutting apart coro shapes then trimming*

Since I had an 8' sheet of coro, I left it full length and just made double-length pieces. This allowed the Icicles to be grouped into segments of  $\approx 300$  pixels each. For longer roof areas where I knew there would be a joint, I tried to match the edges so the seam was not obvious. In some cases I had to extend or stiffen the coro so I slid a hook made of ceiling hanger wire into the corrugations and taped it in place with white duct tape so the wire would stay in place:



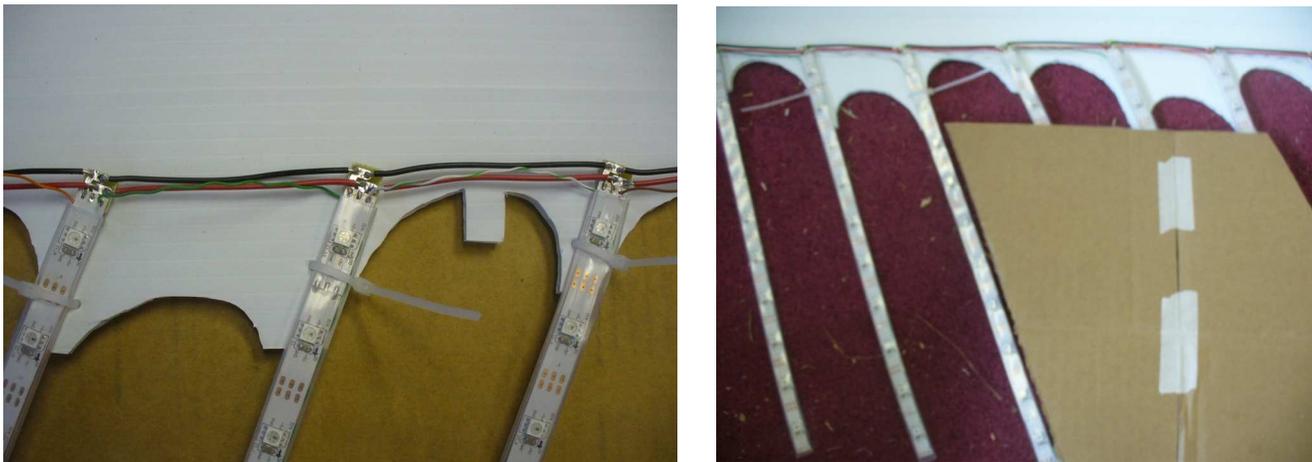
*Cut joints to match; insert wire to strength joints or weak areas*

Due to rafter placement, I did a trial fit of the coro and identified a few places to avoid putting Icicles:



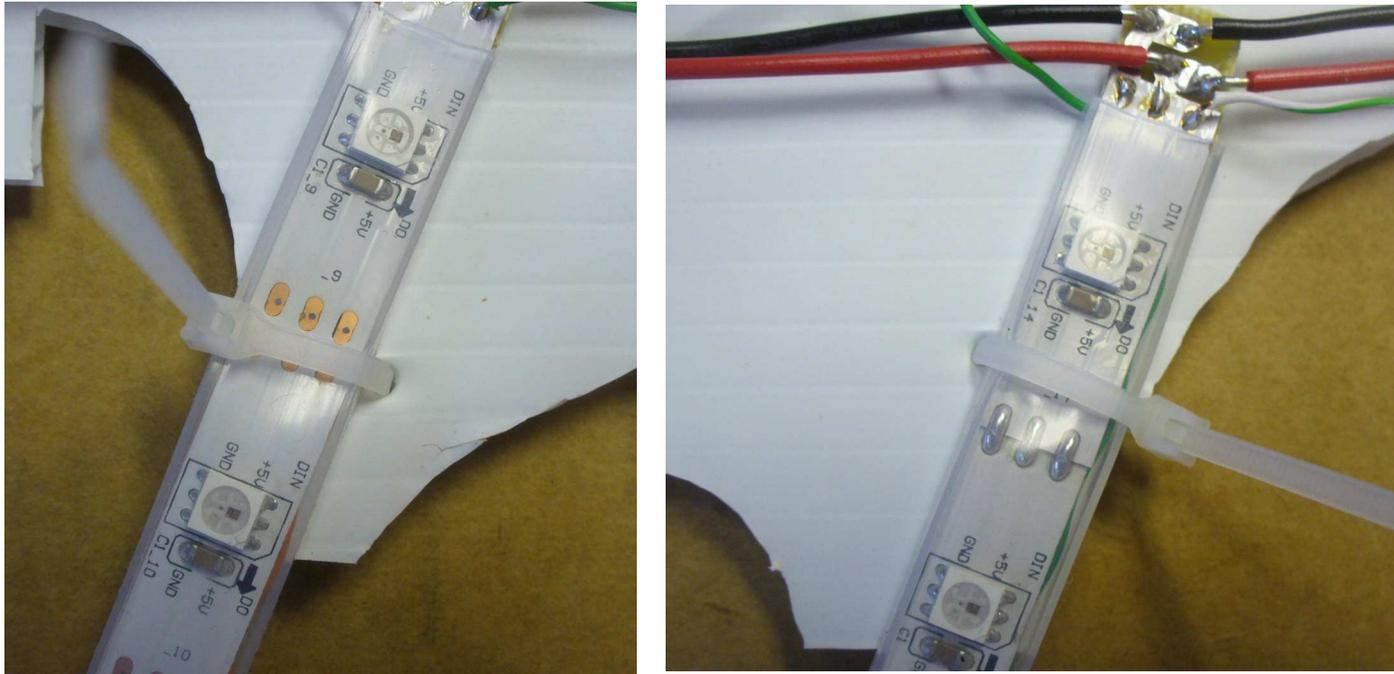
*Coro trial fit, gap to accommodate rafters*

I had left several inches of material above the pattern so it could be folded over the bus wires and Icicle tops to keep the rain out. After cutting, I laid the Icicle assemblies onto the coro and moved them around to determine fastening points. For the slanted Icicles, I had to bend the bus wires to the correct angle, and I used a cardboard guide to get the correct angle:



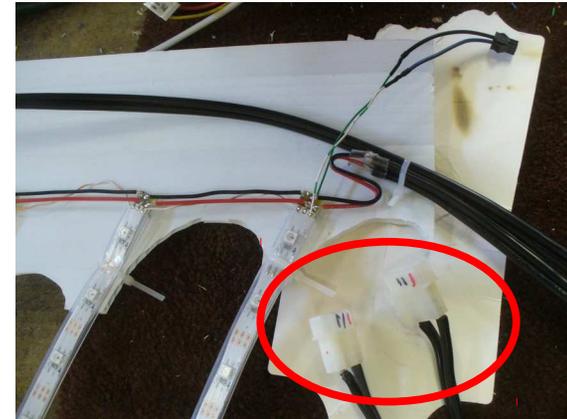
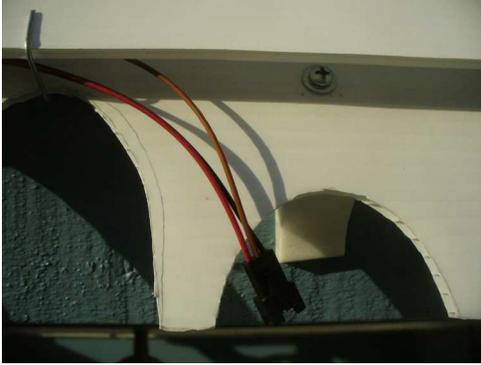
*Icicle placement; cardboard guide for correct angle*

I then used an awl to poke a hole beside each Icicle and zip-tied it to the underlying coro structure. I only used one zip-tie for each Icicle, because the whole group of Icicles on the coro made it reasonably sturdy:



*Poked a hole beside Icicle, zip-tied to coro*

After the Icicles were fastened, I soldered power lines to the bus wires and zip-tied them to the coro back, and also added data in and out connectors so the Icicle sections could be connected to a controller or daisy chained. To limit current and voltage drop in the power lines, I used a separate power connection for each set of 150 Icicles – basically I had 2 separate groups of 15 Icicles contained within each 8' section of coro. At the end of the power leads I added molex connectors to fit with the PC power supply connectors:



*Data in + out, power lines with molex connectors*

After the Icicles and power cords were fastened, I folded the coro “hood” down over the wires and Icicle tops. I cut through one surface to make the folding easier. I also cut notches so the coro hood could lie flat against the coro base, but later I decided that wasn’t necessary and just taped or clipped the hoods down:



*Tape or clip coro hood over Icicle tops and bus wires*

Now for the Bay window, I actually built the coro structure inside out – I mounted the Icicles on the *inside* of the coro and folded the hood area back instead of forward. I also cut holes so the top pixels would show thru:



*Icicles for Bay window mounted on inside, holes cut to expose pixels*

The snow/ice effect with coro looked nice on the Bay window because it was right up at the roof line, but in other places I had mounted the Icicles too far down from the roof shingles so the straight top edge of the coro hood did not look quite right. This is easy to fix by adding another wavy piece of coro above the Icicles, or even a piece of foam padding for fiberglass roofing helps cover the unnaturally straight edge line:



*Top edge of coro looked okay right under shingles, but needed extra material when lower*

For most of the Icicles I folded that extra material at the top forward and over the Icicles. However, for the bay window, I folded it back so it would rest along the inside of the bay overhang.

## 5. Usage

Installing the Icicles took a little extra effort the first year to drill some holes or add hooks, but after that first year the installation was fairly quick – a couple of hours for the entire 80' roof line, and then maybe another hour to install the PC power supplies.

### 5.1. Installation

I had 3 slightly different installation procedures, due to variations in the roof line.

#### 5.1.1. Sloped Roof, Vertical Fascia

Most of the roof line consisted of sloped shingles with a vertical fascia/barge board. For these areas, I just used a washer and self-threading screw to hold the coro back of the Icicles to the barge board:



*Simple washer and screw to fasten coro back of Icicles to fascia/barge board*

During installation, I held the coroplast Icicle assembly horizontally to avoid bending the Icicles or stressing the zip-ties while I climbed the ladder. This posed a bit of a challenge for the 8' sections – the coroplast required support at both ends because it was not strong enough to support its weight when just held up in the middle. I held it with both hands (outstretched arms), but I also needed one hand for the cordless screwdriver. In places where there was a rafter protruding, I rested one end of the coroplast on the rafter while I fastened the other end of the coroplast to the fascia/soffit board. On the higher areas, I attached a wire hook first so it could hold up one end while I fastened the other end. This made it a quick and easy procedure for one person:



*Protruding rafter or hook to hold up one end while fastening other end*

Where adjacent sections of Icicles met, I folded and taped a short piece of coroplast to prevent rain from getting in. For larger gaps (at rafters at roof points), I used a wider piece of coroplast and held it down with a wire clip to prevent rain from getting in to the wires and open Icicle tops:



*Coroplast hood to bridge gaps*

### 5.1.2. Horizontal Roof, Slanted Fascia

For areas of roof line with a slanted fascia/barge board, I used a similar technique but I added spacers behind the coro to make the Icicles vertical:



*Styrofoam wedges between coro and slanted fascia/barge board to make Icicles hang vertically*

As with the sloped Icicles, I bridged gaps with a small coro hood. I also taped over open ends to prevent rain from entering from the side:



*Coro hood to cover gaps, tape over open ends*

### 5.1.3. Bay Window

The bay window had no fascia/barge boards, just an overhang so I folded the coro back and hooked it under some old Christmas light clips. At corners I cut through the outer layer of coro to allow a bend to fit the window:



*Coro folded back on Bay window Icicles, hooked under old Christmas light clips*

This hid the wires but allowed easy access from below. I also used a wire hook at corners to secure the Icicles to the overhang:



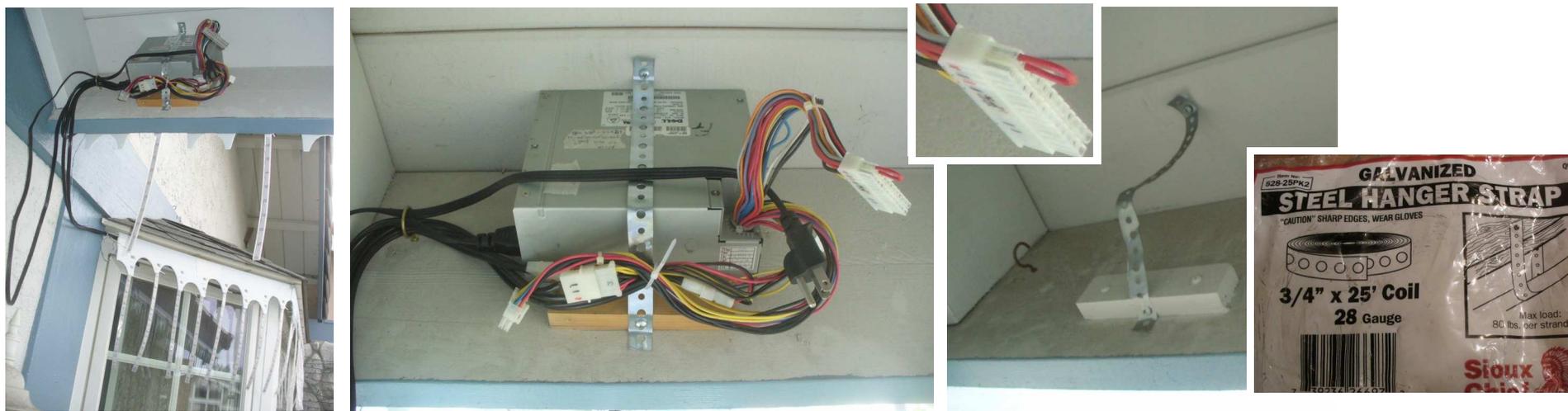
*Easy access to wires from below, wire hook secures Icicles to Bay window overhang*

## 5.2. Power Supplies

I salvaged some PC power supplies that were in good working condition, so I used those to power the Icicles. I've found PC power supplies to be very convenient because they are readily available, and they can power either 5V or 12V pixels<sup>14</sup>. This gives me the freedom to switch between 5V and 12V pixels as needed, either during the planning phase or for emergency repairs during show season.

I mounted the power supplies up under the eaves to protect them from rain. This allowed me to keep the power leads short so I didn't run into voltage drop issues with 5V pixels. The power supplies are a little bulkier and can look messy with all the wires hanging out (unless you trim them), but this wasn't a problem since they were hidden up under the eaves. I also have some new dedicated 5V pixel power supplies, which I use more for when space or fewer wires is important.

I take down the power supplies when they are not in use during the off season, so I wanted a quick and easy way to install and uninstall them. I permanently fastened a little block of wood to help support the power supplies, and then I attached a loop of earthquake strapping. This can be tightened or loosened as needed to hold the power supplies securely but allow easy removal:



*PC power supply mounted in eaves, held in with earthquake strapping but easily removed*

14 Just don't power other devices such as FM transmitters from the same power supply as pixels – too much noise and interference!

Most of the PC power supplies I have are rated for 20A or more on the 5V line, which allows them to drive 300+ pixels easily. That corresponds nicely to the pixel count on each 8' section of Icicles. I don't use full-white in the sequences, but I've recently started limiting full white to 83% in the software to prevent accidentally running the power supplies too close to their rated limit. This also cuts down on power usage.

In some areas, I used 2 power supplies between sections of Icicles (here, a 60A dedicated 5V pixel power supply would have been more compact):



*Multiple power supplies hidden up under eaves, earthquake strap for easy removal*

### 5.3. Data Lines

For the first few years I used my existing Renard controllers (a mix of custom<sup>15</sup> and PX-1<sup>16</sup> boards) to control the Icicles, running custom firmware to handle the WS281X pixels. I used one twisted pair of cat5 to carry data from each PIC chip to a section of Icicles (data on one wire and common ground on the other). This resulted in a total of 6 twisted pairs going out to the Icicles - one PIC16F688 to drive each 8' Icicle section of 300 or so pixels<sup>17</sup>, and the PIC16F1825 on the PX-1 to drive a larger section of  $\approx 500$  pixels, for a total of  $\approx 2100$  pixels.

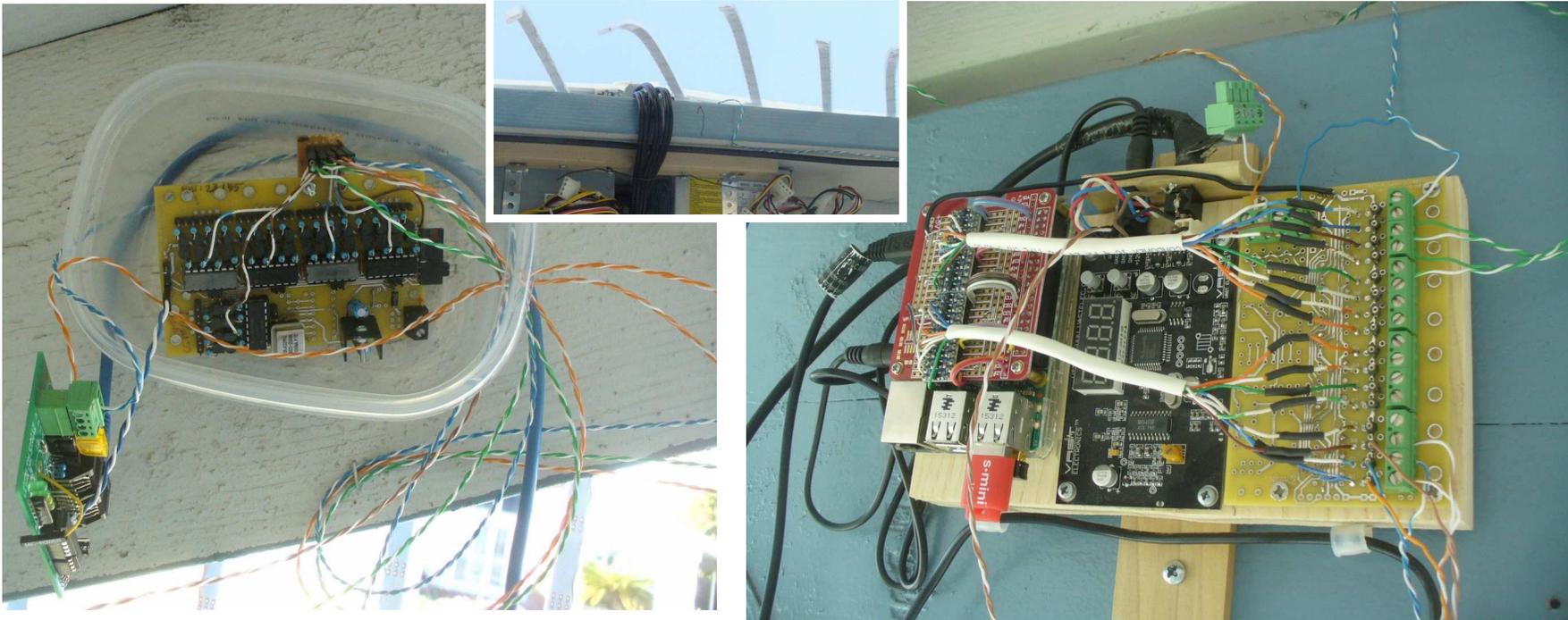
15 Equivalent to the circuit shown for <https://www.doityourselfchristmas.com/wiki/index.php?title=Renard>

16 Renard PX-1 described at [https://www.doityourselfchristmas.com/wiki/index.php?title=Renard\\_PX1\\_Pixel\\_Controller](https://www.doityourselfchristmas.com/wiki/index.php?title=Renard_PX1_Pixel_Controller)

17 My custom firmware handled up to 384 pixels per 16F688.

This past season I switched over to using a RPi for everything. I daisy chained multiple Icicle sections together with about  $\approx 1K$  pixels in each and set my “universe” size to 1150<sup>18</sup>, which allowed me to consolidate the 6 data lines down to 2.

In both cases, I placed the controllers up under the eaves and just strung twisted pairs from there out to the Icicles:



*Cat5 twisted pair carries data from Renard or RPi to each Icicle section*

## 5.4. Sequencing

I’m not going to say much about sequencing or effects here, other than just to point out that the Icicles can be represented by a rectangular grid. I mostly wired my Icicles right-to-left for some reason, so I used a right-to-left, top-to-bottom grid. As long as your sequencing software can accommodate custom models it should work.

---

<sup>18</sup> More custom software

I used a spreadsheet like Microsoft Excel or OpenOffice Calc to create the model first so I could renumber the pixels more easily, then I copied the node numbers into xLights as a custom model. That was a whole lot easier than trying to enter 1000s of node numbers directly into xLights.

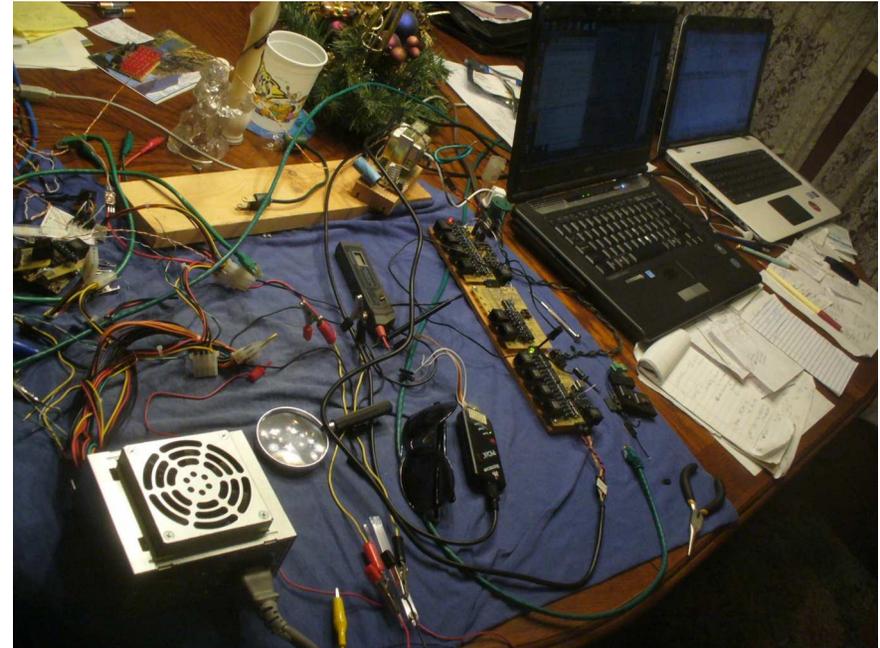
The screenshot shows a spreadsheet with the following data points:

- Row 1:** Segs: 207, 7 segment: 33, 30, 30, 32, 34, 15, 33, 2 ports: 93, 114
- Row 2:** Amps: 124, 19.8, 18, 18, 19.2, 20.4, 9, 10.8, 55.8, 68.4
- Row 3:** %80%: 99.4, 15.84, 14.4, 14.4, 15.36, 16.32, 7.2, 15.84, 44.64, 54.72
- Row 5:** NEW all in one: #/ctr: 384
- Row 18:** total padding: 276 #pad 4, total icicles: 2070 vs. 2070
- Row 20:** OLD: R/C 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29
- Row 21:** R/C 9 19 29 39 49 59 69 79 89 99 109 119 129 139 149 159 169 179 189 199 209 219 229 239 249 259 269 279 289 299
- Row 22:** R/C 8 18 28 38 48 58 68 78 88 98 108 118 128 138 148 158 168 178 188 198 208 218 228 238 248 258 268 278 288 298
- Row 23:** R/C 7 17 27 37 47 57 67 77 87 97 107 117 127 137 147 157 167 177 187 197 207 217 227 237 247 257 267 277 287 297
- Row 24:** R/C 6 16 26 36 46 56 66 76 86 96 106 116 126 136 146 156 166 176 186 196 206 216 226 236 246 256 266 276 286 296
- Row 25:** R/C 5 15 25 35 45 55 65 75 85 95 105 115 125 135 145 155 165 175 185 195 205 215 225 235 245 255 265 275 285 295
- Row 26:** R/C 4 14 24 34 44 54 64 74 84 94 104 114 124 134 144 154 164 174 184 194 204 214 224 234 244 254 264 274 284 294
- Row 27:** R/C 3 13 23 33 43 53 63 73 83 93 103 113 123 133 143 153 163 173 183 193 203 213 223 233 243 253 263 273 283 293
- Row 28:** R/C 2 12 22 32 42 52 62 72 82 92 102 112 122 132 142 152 162 172 182 192 202 212 222 232 242 252 262 272 282 292
- Row 29:** R/C 1 11 21 31 41 51 61 71 81 91 101 111 121 131 141 151 161 171 181 191 201 211 221 231 241 251 261 271 281 291
- Row 30:** R/C 0 10 20 30 40 50 60 70 80 90 100 110 120 130 140 150 160 170 180 190 200 210 220 230 240 250 260 270 280 290
- Row 31:** R/C 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29
- Row 32:** R/C 309 319 329 339 349 359 369 379 389 399 409 419 429 439 449 459 469 479 489 499 509 519 529 539 549 559 569 579 589 599
- Row 33:** R/C 308 318 328 338 348 358 368 378 388 398 408 418 428 438 448 458 468 478 488 498 508 518 528 538 548 558 568 578 588 598
- Row 34:** R/C 307 317 327 337 347 357 367 377 387 397 407 417 427 437 447 457 467 477 487 497 507 517 527 537 547 557 567 577 587 597
- Row 35:** R/C 306 316 326 336 346 356 366 376 386 396 406 416 426 436 446 456 466 476 486 496 506 516 526 536 546 556 566 576 586 596

Partial model of Icicles in a spreadsheet

## 5.5. Storage

During the off season I store the Icicles flat on a shelf in the ceiling of the garage. They don't take up much space, as can be seen here on this temporary table:



*Icicles store flat, don't take up much space; now back to the lab ☺*

## 6. More Info

If you have any questions or comments for improvement of this document, please send email to [techguy@eShepherdsOfLight.com](mailto:techguy@eShepherdsOfLight.com).

Version	Description	Date
1.0	First version, finally	2/20/17

Copyright © eShepherdsOfLight.com

May be freely distributed, copied, and used for non-commercial purposes.

-eof-