Water Drip Dress with Oozemaster 3000

Created by Erin St Blaine


Last updated on 2023-08-29 04:57:32 PM EDT
# Table of Contents

## Overview
- Controller: Feather SCORPIO
- NeoPixels
- Power
- Wire & Accessories
- Additional Materials

## Planning
- Dress Selection
- Pixel Layout Planning
- Data Flow
- Dress Build Planning
- Enlist Help

## Arduino Code
- Code Modifications

## Wiring Diagram

## Feather Scorpio Wiring

## NeoPixel Strip Wiring
- Wire Management
- "Ooze" Sections: Side-Light Pixels
- "Drip" Sections: 60/m NeoPixels
- "Splat" Pixels: Slim NeoPixel Dots
- Sealing

## Costume Build
Overview

Create a costume that oozes charm and sophistication, and shows you off like the Sea Goddess you are.

This is a fairly advanced project, with some costuming skills and a lot of work with wiring and soldering. If you don't already have mad soldering skills, you will have them by the time you're done with this project.

This is also not a step-by-step build guide. I'll go over tips and techniques and show you the path I took and the mistakes I made, to give you the tools to go and build your own magical creation.
Controller: Feather SCORPIO

Adafruit Feather RP2040 SCORPIO - 8 Channel NeoPixel Driver
If there is one thing Adafruit is known for, its mega-blinky-fun-rainbow-LEDs. We just love sticking NeoPixels anywhere and...
https://www.adafruit.com/product/5650

This board is designed to drive multiple pixel strands at the same time. It has 5v logic level shifting built-in as well as some other nice features like onboard battery charging and LOTS of GPIO pins. It's perfect for a project like this that's using some heavyweight code and lots of pixel strands.

It has an individual ground pin for each of the strands we'll be using, and the ground and data pins are present along one end of the board to make it easy to get all those wires attached.

NeoPixels

I used three different types of pixels for this dress: 90/m side-light pixels for the top / "ooze" sections, 60/m standard pixels for the long drop sections, and 20/m led "dots" for the splats along the hem.

Adafruit NeoPixel LED Side Light Strip - Black 90 LED
Fancy new side light LED strips are a great alternative for folks who have loved and used Adafruit LED strips for a few years but want gorgeous, glowy light emitting at...
https://www.adafruit.com/product/3635
Adafruit NeoPixel Digital RGB LED Strip - White 60 LED
You thought it couldn't get better than our world-famous 32-LED-per-meter Digital LED strip but we will prove you wrong! You wanted...
https://www.adafruit.com/product/1138

Adafruit NeoPixel Slim LED Dot Strand - 20 LEDs at 2" Pitch
We have all sorts of LED strips for a wide range of needs. Chonky strips? We got those!
https://www.adafruit.com/product/5225

Construction was simplified using a full strand of “dots” along the hem, though just a subset of pixels are actually used. If you’re patient with some extra wiring, soldering and strain relief, individual FLORA NeoPixels — one per drip — could be used instead.

Power

There are several options for powering this project. I am powering via the Feather’s onboard USB port using a USB battery pack. We have a couple great ones in the shop, and there are many more shapes and sizes available online.

These don't have an on/off switch included, but you can plug/unplug to turn the costume on or off, or find a USB C cable with an inline power switch for easy on/off.

<table>
<thead>
<tr>
<th>1 x USB Battery Pack - 10000mAh</th>
<th>1 x USB Battery Pack - 2200mAh</th>
</tr>
</thead>
<tbody>
<tr>
<td>USB Battery Pack 10000mAh - 2 x 5V outputs</td>
<td>USB Battery Pack - 2200 mAh Capacity - 5V 1A Output</td>
</tr>
</tbody>
</table>
You could also power through the onboard JST connector using a lithium-polymer battery. LiPoly cells are small and efficient but come with safety concerns, and aren't always the best option for a costume, especially if it will be worn by a kid or if it might get wet or damaged.

1 x Lithium Polymer Battery - 2500mAh  
https://www.adafruit.com/product/328
Lithium Ion Polymer Battery - 3.7v 2500mAh

## Wire & Accessories

I used a LOT of wire for this project, and a lot of heat shrink to cover up all those wire connections. Since this is a wearable, I wanted to keep the loose wires to a minimum, so I used 28awg Silicon Stranded Ribbon Cable. This way most of the longer wire runs have 3 wires that are stuck together, giving me effectively one wire to deal with instead of three.

I estimated 1m for each of my strands, plus a little extra, to give myself plenty of wiggle room. I also used some single-strand red silicone stranded wire for the power connections.

10 x Silicone Stranded Ribbon Cable  
Silicone Cover Stranded-Core Ribbon Cable - 4 Wires 1 Meter Long - 28AWG Black  
https://www.adafruit.com/product/3891

1 x Red Silicone Stranded Wire  
Silicone Cover Stranded-Core Wire - 25ft 26AWG - Red  
https://www.adafruit.com/product/2513

You'll do a lot of wire splicing so be sure you've got enough heat shrink. It's also really helpful to have large clear heat shrink for sealing the ends of your LED strips.

1 x Heat Shrink Pack  
Heat Shrink Pack  
https://www.adafruit.com/product/344

3 x Clear Heat Shrink  
Food-Grade Heat Shrink - 3/8" diameter 12" long  
https://www.adafruit.com/product/1020

If you want to add "Carrie Mode" where the lights all turn red, you'll need a momentary switch button.

1 x Tactile Switch Buttons  
Tactile Switch Buttons (12mm square, 6mm tall) x 10 pack  
https://www.adafruit.com/product/1119
Depending on your build, it could be useful to have a few 3-pin JST connectors. I used these to plug and unplug the drips in the back, to minimize pulling on wires when I pull the dress on and off.

**3 x JST Connectors**
3-pin JST SM Plug + Receptacle Cable Set

https://www.adafruit.com/product/1663

### Additional Materials

- A fabulous dress or cloak or costume
- A sturdy slip, or some fabric for your under-layer (optional)
- Short USB C cable () and/or power switch cable () for your battery
- Needle and heavy-duty thread

### Planning

![Image of a person in a dress](image)

### Dress Selection

I chose a [white dress with sequin fabric and a thin lining from Hebeos.com ()](https://www.hebeos.com). Lighter colors will show the light more brightly, darker colors will make a more subtle and less diffused end result. It's not a bad idea to bring a pixel strand with you when you shop for your dress, and try holding it behind different fabrics to find the result you want.

My dress is form-fitting but not tight. A skintight dress will not hide the wires and pixel strips as well as one with a little bit of ease will. I also chose a velvet fabric since it's a little thicker and will do a good job of hiding the electronics inside.
Pixel Layout Planning

Figure out how many drips you want. My dress has seven drips: three across the front, one on each side seam, and two on the back. The Oozemaster code will run up to 7 individual drips -- if you want more drips, you can either build two separate, unconnected systems or you can mirror some of the drips. Check out this page for more info. ()

The original Ooze Master project has one strip per drip. This project takes an unconventional turn — each drip is “Y” shaped — where two streams converge at a point before falling. This is done with clever NeoPixel wiring and doesn’t require anything special in the code.

Figure out how many pixels go in each drip. Each drip has three sections:

1. The top "ooze" section, made of two lengths 90/m side-light NeoPixel strips; one length will be 1 pixel shorter than the other, e.g. 9 and 8 pixels.
2. The vertical "drip" section, made of 60/m pixels.
3. The bottom "splat" section, made of slim NeoPixel dots.

My dress used

• 120 pixels of the 90/m side-light NeoPixels (1.5 m)
• 136 pixels of the 60/m standard NeoPixels (a little over 2 m).
• 20 pixels of the 20/m slim NeoPixel dots (one strand)

The drip action looks best if the vertical "drip" portion spans about half the distance to the "splat" pixel on the hem.

Keep in mind that you'll want to be able to sit down in this dress without breaking your LED strips. The pixels bend nicely front-to-back but will break if you bend them side-to-side. I stopped the pixel strips above my hips so that sitting down won't break any of the pixel strips.

Data Flow

This photo shows the data flow direction of the strips. The red arrows are 90/m side light pixels, and the two halves are mirrored so they're showing the exact same thing. The green arrows are 60/m NeoPixels and the orange arrows are the slim NeoPixel dots.

I ended up adding strips on each side under the armpits as well. And for my final version I found the splats looked best when they stayed on the hem, rather than going up the slit in the dress.
Dress Build Planning

My dress is white. And it drags on the ground. It's going to need washing every time I wear it.

It is possible to wash a dress with electronics in it but it's really pretty hard on the electronics. Also, LED strips break no matter how robust I try and make them. It seems to be pretty unavoidable. If this was easy, everyone would be doing it.

In order to make repairs and maintenance as easy as possible, I decided to attach the lights and electronics to an underskirt instead of sewing them to my fancy dress. This way I can wash the dress as needed, and I also have very easy access to the lights without having to cut apart my dress lining.

Enlist Help

Though it would seem a simple thing, it's a Costuming 101 fact that measuring or placing things on your own body is unreliable and error-prone. To position and align these LED strips, you'll either need a “dress form” that's sized or adjustable to your body measurements, or a trusted friend who understands the project and can help measure and mark while you wear the base garment.

Arduino Code

This project uses the Oozemaster 3000 code by Phil Burgess. There's a fantastic step-by-step guide to getting the code installed here:

Ooze Master 3000 Tutorial Arduino Code

Since we're using the Feather RP2040 Scorpio, there's no need to change pin numbers -- this code will work immediately as installed.
Get your software up and running first so you'll be able to see your strips working as you build. Then, come back to this page to tweak the code and make it fit to your project.

**Code Modifications**

Once you've completed your build, come on back here to make the necessary tweaks and modifications to make the code work with your build.

**Pixel Pitch**

Near the top of the code is this line:

```
#define PIXEL_PITCH (1.0 / 150.0) // 150 pixels/m
```

This tells the code the “density” of your NeoPixel strip, in LEDs-per-meter. The default, 150.0, corresponds to the density of the ultra-skinny NeoPixel strip. The code needs to know this figure so the drippings fall at a physically plausible speed.

We are mixing and matching strip densities, so some experimentation is required. I found that the most important part of the illusion is the long vertical drip, so I set this at 60/m since that's the density of those strips. It makes the "ooze" at the top a bit slower, but I actually like the way that looks. So 60/m it is.

**Colors**

The Oozemaster code has been updated to allow for multiple colors on the strips. To set up your colors, look for this section:

```
uint8_t palette[][3] = {
    { 0, 255, 0 }, // Bright green ectoplasm
};
```

As explained on this page, you can set up as many colors as you like and the code will choose randomly between them. For my dress, I wanted a subtle variety of pastel colors in the "cool" spectrum -- pale greens and blues that evoke a subtly changing waterfall. I also wanted some brightness variation so some drips are brighter than others. Here's the palette I settled on:

```
uint8_t palette[][3] = {
    { 50, 50, 50 }, // Color 0
    { 60, 85, 95 }, // Color 1
    { 70, 90, 100 }, // Color 2
};
```
Pixel Counts for Each Section

Look for this section a bit further down. This is where you'll tell the code how many pixels you have in each strip, and where you'd like the "ooze" to stop, the "drip" to start, and also which color(s) you want to use from the palette you just set up.

The first number is the total number of pixels in the section. The second is the pixel you'd like the "pause" to happen on. The third number is the distance from the "pause" point to the "splat" pixel on the ground, and the fourth and fifth numbers determine which color(s) each drip will use. Here's my modified code:

```cpp
} drip[] = {
    // THIS TABLE CONTAINS INFO FOR UP TO 8 NEOPixel DRIPS
    { 44,  9, 1.143, 0, 7 }, // Left front, NeoPXL8 output 0: 44 pixels long, pause at index 9, 1.143 meters up
    { 57, 23, 1.244, 0, 7 }, // Middle front, NeoPXL8 output 1: 57 pixels long, drip pauses at index 23, 1.244 meters above ground, use palette colors 0-7
    { 44,  9, 1.143, 0, 7 }, // Right Front NeoPXL8 output 2: etc.
    { 21,  8, 0.96 , 0, 7 }, // Right Back NeoPXL8 output 3
    { 21,  8, 0.96 , 0, 7 }, // Left Back NeoPXL8 output 4
    { 30,  8, 1.193 , 0, 7 }, // Right Side NeoPXL8 output 5
    { 30,  6, 1.193, 0, 7 }, // Left Side NeoPXL8 output 6
    // NeoPXL8 output 7 is normally reserved for ground splats
    // You CAN add an eighth drip here, but then will not get splats
};
```

Note that we're using only colors 0–7 since I don't want the red (#8) to appear randomly. The red color will be used for Carrie Mode: the drips will change to this color when I press the button we've wired to A2-A3. I chose a blood-red color, about half brightness with a little bit of blue mixed in to keep it from looking overly happy and bright.

If you're using Carrie Mode, also be sure to un-comment the two lines near the top and be sure your switch pin numbers match your wiring:

```cpp
#define CARRIE_PIN    A2
#define CARRIE_GROUND A3
```

©Adafruit Industries
Splat Map

To make our wiring easier, the code gives us the ability to sew in a whole NeoPixel strand and then choose which pixel lights up in association with each drip. Look for this line:

```c
uint8_t splatmap[] = { 1, 19, 17, 7, 11, 4, 15 };
```

I hung my dress on a mannequin and counted out the pixels, choosing the one on the "splat" strip that best aligned with each "drip". Change this line to assign the correct pixel to each drip, in the order the drips are soldered to the board.

Splat Fade Timing

The "splat" pixels at the bottom are not quite as obvious as I'd like with the code as-written -- they disappear too fast. If you want them to hang around a little longer, look for this around line 239 in the code:

```c
drip[i].splatDurationUsec = random(900000, 1100000);
```

This chooses a random number between 90-110 milliseconds, after which the drip will fade. To keep it on longer, you can change these numbers to something bigger. I wanted them to stay on for between 6-8 seconds so I changed the line to read:

```c
drip[i].splatDurationUsec = random(6000000, 8000000);
```
The wiring for this project is a bit complicated, but the **Feather RP2040 Scorpio** makes it as easy as possible to get all these strips wired up and running.

For my project I've got seven strips running drips: three in the front, one on each side seam, and two on the back. The 8th strip is used for the splats around the hem of the dress. The diagram above is a conceptual schematic; the actual strip lengths will be different, and will depend on the lengths you decide.

The seven drip strips are each split into two sections: the top "ooze" section uses two identical strips of **90/m side-light NeoPixels**, which then feed into a longer section of **60/m standard NeoPixels** for the vertical "drip" section. The "splat" strip around the hemline is a strand of **Slim NeoPixel Dots**.

Each pixel strip is connected to its own data and ground pin, which are helpfully arranged along the bottom of the Feather. The 5V+ connections from the NeoPixels all get wired together and connected to USB (if you're powering with a USB battery or through the USB port) or BAT (if you're powering with a LiPoly battery).

There is a momentary switch button wired into pins A2 and A3. This button is used to activate "Carrie Mode" which temporarily turns all the drips to blood red.
Feather Scorpio Wiring

If you're powering with a USB battery, solder a red power wire to USB. If you're using the JST connector battery port for a LiPoly battery, solder to BAT instead.

Split the power wire, then split each wire twice more so you have 8 connections total, one for each LED strip.

Wire your momentary switch to pins A2 and A3. It doesn't matter which leg goes to which pin.

Cover your connections with 3/4" heat shrink and fill in the gap with hot glue. This button will probably get tugged on a lot and we need to make it bulletproof.

Finally, we'll need to attach each of the pixel strip sections to the Feather. I started by making the strips, then pinned them into place before trimming the wires to minimize extra wire in the project. The seven drip strips attach to pins 0-6, and the 8th "ooze" strip attaches to pin 7. The pins each have their own corresponding Ground pin located just below each data pin.

The third 5v wire from each strip will attach to one of the power wires in the "squid" you just made.
NeoPixel Strip Wiring

The majority of the work for this project is getting the pixel strips soldered. As we go, we also really want to minimize the amount of loose wire hanging around. Wires can get pulled or tugged out of place really easily, so we'll spend a lot of time managing wires and sealing strips.

If you're new to soldering NeoPixels (or could use some tips on making it easier), check out our Make it Glow: How to Solder NeoPixels guide.

It is essential to test each strip connection as you go, and test again (and again) as you build out your costume. I like to keep a Circuit Playground Express loaded with test code and some alligator clips on hand. This makes it much easier to chase down any cold solder joints or crossed wires as you're building out your costume.

Wire Management

One way to keep your wires tidy is to use silicone stranded ribbon cable. This 28awg (American Wire Gauge) wire is thin and flexible, and will not break when it's flexed again and again.

I only use silicone stranded wire in my wearable projects. Non-silicone wire will break the 10th or 11th time you wear your project. No bueno. Seriously, spend the extra dollar and use good wire.

This ribbon cable has 4 wires, with a striped wire on one edge. We only need 3 wires, so pull off the 4th wire -- the one opposite edge from the striped wire.
I minimized the amount of loose wire by sliding the soldered strips out of the casing, soldering, then sliding the casing back over both the strip and the wire. It was a little fiddly to get it all in there, but it does fit when using 28awg ribbon cable, and will definitely minimize breakage down the line.

"Ooze" Sections: Side-Light Pixels

I'm using side-light strips for the top "ooze" sections because I want a diffused look with a high pixel density. If the pixels are shining right at the viewer, each individual pixel is really visible. When they're indirect and shining down on the fabric, the viewer sees the reflected light rather than the direct light from the pixels. This makes the dress look like it's glowing magically rather than full of wires and technology.

More about pixel diffusion here ()

These strips are available in 60/m, 90/m or 120/m densities. I found the 90/m to be dense enough to give me smooth buttery animation without overloading the project with hundreds of lights.

For the double-sided "ooze" portions (along the neckline and under the bust line) I cut two strips, one for each side of the "vee" shape. I soldered the longer side up to a long ribbon cable wire, then spliced in a wire that was long enough to reach the far end of the vee. The "drip" portion will get soldered to the "out" end of the longer strip.
Once the pixels are correctly soldered and tested, slide the silicone sheath back over both the strip and the wire to keep it managed and out of the way.

Since the data is coming from different directions, one side of your vee will need to be mounted face-down in order for all the lights to be facing the same way. No problem: the strips look the same whether they're mounted face-up or face-down.

For the Neckline
The side-light strips don't bend easily around a curved neckline. In order to get them to fit the curves, I cut the pixels in the middle of my strip and soldered about an inch of ribbon cable between the two halves. Then I re-sealed the strip using clear heat shrink and hot glue with the wire stuffed inside the silicone casing. This gave me a nice bend that follows my neckline much better.

"Drip" Sections: 60/m NeoPixels

I used regular 60/m NeoPixels for my longer "drip" sections. I did some testing, and the lights move so quickly during the "drip" that a medium-density pixel strip still looks fantastic. The code is written for a single density of strip, but I found that mixing 60/m and 90/m pixels still looks completely realistic.
There are a few benefits to using these medium-density strips rather than the delicate 90/m side-light strips or the pricey 144/m strips that are available.

- The "drip" sections make up the majority of the pixels in this project. If I use 144/m it will almost double my power usage and also nearly double the cost of the lights.
- These strips will get flexed the most, since they're crossing my waistline. I want strips that are a little tougher and that will be able to stand up to being bent or sat upon. The side-light pixels are really breakable when flexed, and I've found that these 60/m pixels really do hold up better in costumes.

Solder about 1" of wire to the "out" end of the longer side of your ooze section vee. Connect the "in" end of your drip strip.

Once each section is assembled, and the wires are managed as neatly as possible, test your strip again. It's easy to pull wires out while you're futzing around. It's much easier to fix any broken joints now than it will be when the pixels are on the dress.

"Splat" Pixels: Slim NeoPixel Dots

The little drips where the lights appear to hit the ground are maybe the coolest part of this effect. They're also the most vulnerable: any lights in your hemline are going to get stretched, kicked, stepped on and tripped over. In my design, they also need to be sewn into the dress itself instead of attached to an underlayer -- though you could add them to a slip or hoop skirt depending on your build. My dress has a slit up the
front so doesn't lend itself to undergarments, so I decided to attach them permanently.

I used one strand of Slim NeoPixel Dots (1) for this part of the project. These are really tough LEDs. Each pixel is encased in resin with super flexy wires in between. I haven't managed to break one of these strands yet. I can also leave them in the dress for washing -- a little dunking isn't going to hurt them as long as you let them dry completely before powering again.

The "splat" strand attaches to the two rightmost pins on the bottom of the controller (as well as a power wire). I soldered on a long ribbon cable wire and put a 3-pin connector on the end so I can easily detach the the dress from the underdress.

We will be able to select which LED in this strand is associated with which drip in the code, so I didn't worry too much about exact placement. I just made sure there was some slack between the pixels to minimize any pulling.

I made a little slit in the fabric and slid the whole strip inside, then sewed it into place by feel through the lining of the dress. I didn't even have to detach the lining, I was able to get all the lights in place by pushing them around through the fabric.

I plan to change this arrangement a bit because the wire from the connector shows when I wear the dress. I will add another 3-pin connector up near the microcontroller and thread the whole wire down inside the lining, so I'll be able to plug it in someplace less obvious.
Sealing

Once you're 100% sure all your strips are working, and your wires are as managed as you can get them, seal up the ends of all your strips by sliding a small piece of clear 3/4" or 1/2" heat shrink over the end of the strip, covering the wire connections. Fill the heat shrink up with hot glue, then use a heat gun to shrink it down while the glue is still wet. When the glue dries, your connections will be potted in glue and covered in plastic, and are much, MUCH less likely to break when they get pulled on.

Seal every single connection before attaching it to your costume.
Costume Build

LED costumes can be a challenge to maintain, especially ones that will need to be washed fairly often. To protect my electronics and make the dress easy to fix or update, I put almost everything onto an under-dress.

I made this out of stretch cotton fabric so it hugs my body and doesn't create too much bulk under the dress. It would also work well to put the lights on the outside of a corset or a slip.

To get the perfect lines, I turned my dress inside out and traced it onto the fabric, transferring the darts and seam lines. My dress is a pretty simple pattern so this wasn't too hard.

Pulling this on over my head in a hurry would be a great way to break all the wires, so I added a side zipper to my under-dress so I can put it on like an apron. This will minimize breakage a lot.
I didn't want to run the wires all the way around my body, so added a couple of connectors that go across the zipper to plug in my battery and the strips on the back.

In hindsight, I realize that if I'd put the zipper on the other side seam I could have avoided this problem.

I added a battery pocket on the center back, near the top so it can be changed while I'm wearing the dress without too much fuss. I also added an on/off switch between the battery and the rest of the project using a short USB C cable with a switch inline.

I also encased the Feather inside a piece of clear 1/2" heat shrink tubing. This will protect it from snags and from sweat.

The fourth connector goes to the "splat" lights in the hem of the dress. I added a long wire to the LED strand with a connector at the end that runs up between the dress lining and the fabric, so it's hidden and the connector emerges in the right spot.

I took some time to stitch down ALL the components. It's easy for wires that are sticking out to get caught on something and pull out, usually right before you’re ready to go onstage or walk down the aisle. A stitch in time can save your whole night.