instructables Dynamic Neon Arduino Driven Sign Instructions

Contents

instructables

Instructables Dynamic Neon Arduino Driven Sign

Product Information Dynamic Neon Arduino Driven Sign

The Dynamic Neon Arduino Driven Sign is a DIY LED sign that can display various groovy patterns. The sign is made using LED neon strips, Arduino Uno microcontroller board, an NPN transistor, a terminal block, toggle switch, sheet wood, screws, and a 12V DC power supply. The sign can be used to display any kind of lettering for events, shops, or homes.

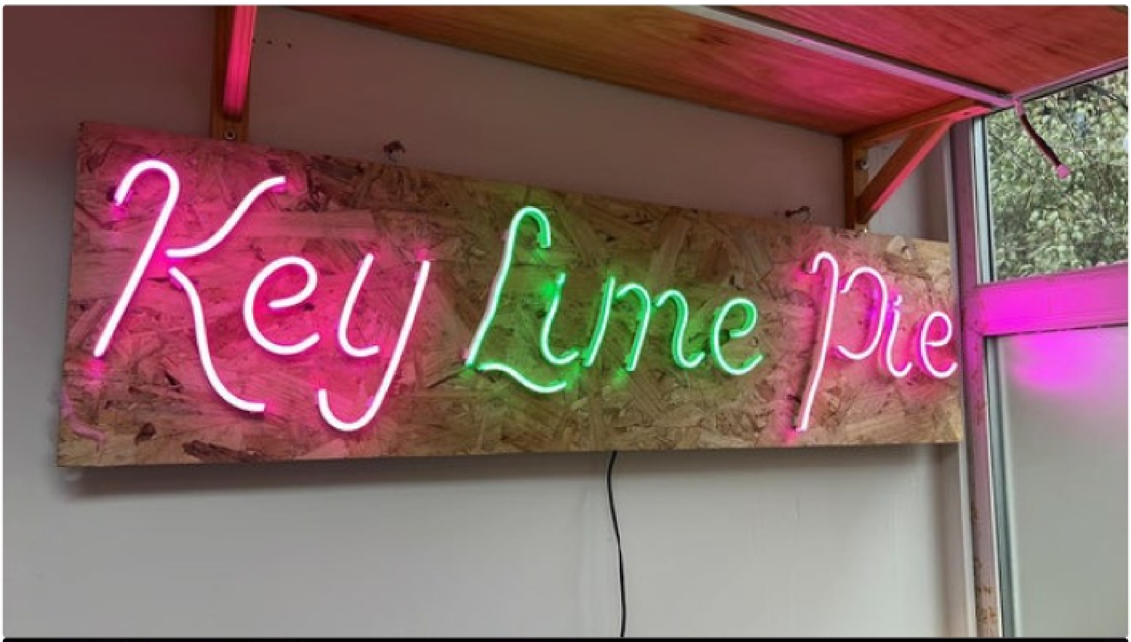
Supplies

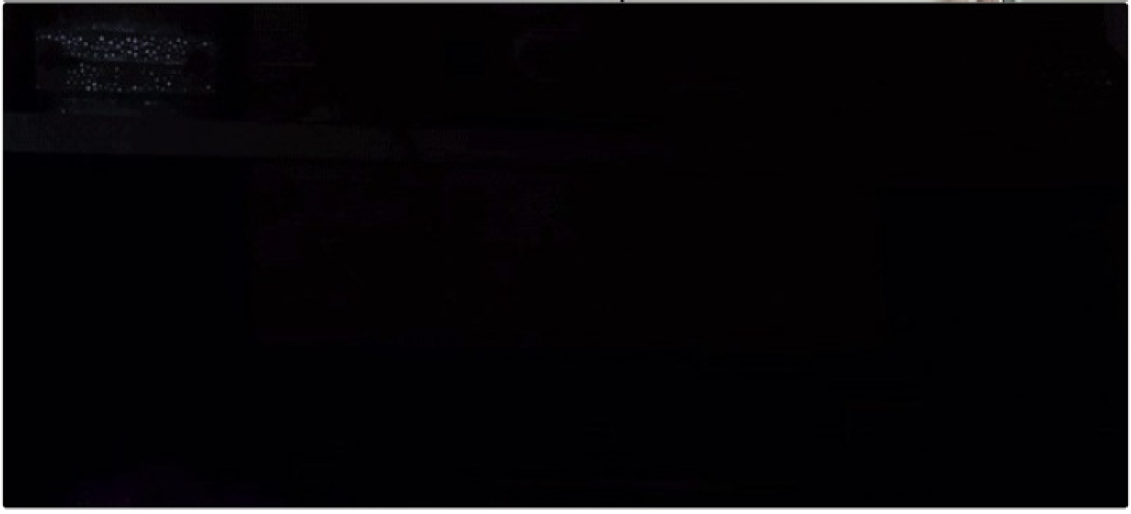
- LED Neon strip (Amazon/Ebay)
- Sheet wood
- Screws
- Arduino Uno
- BC639 (or any suitable NPN transistor)
- Terminal block
- Toggle Switch
- Double multi-stranded wire
- 12V DC power supply
- Soldering Iron

Optional

- Projector
- 3D Printer
- Dog

Product Usage Instructions



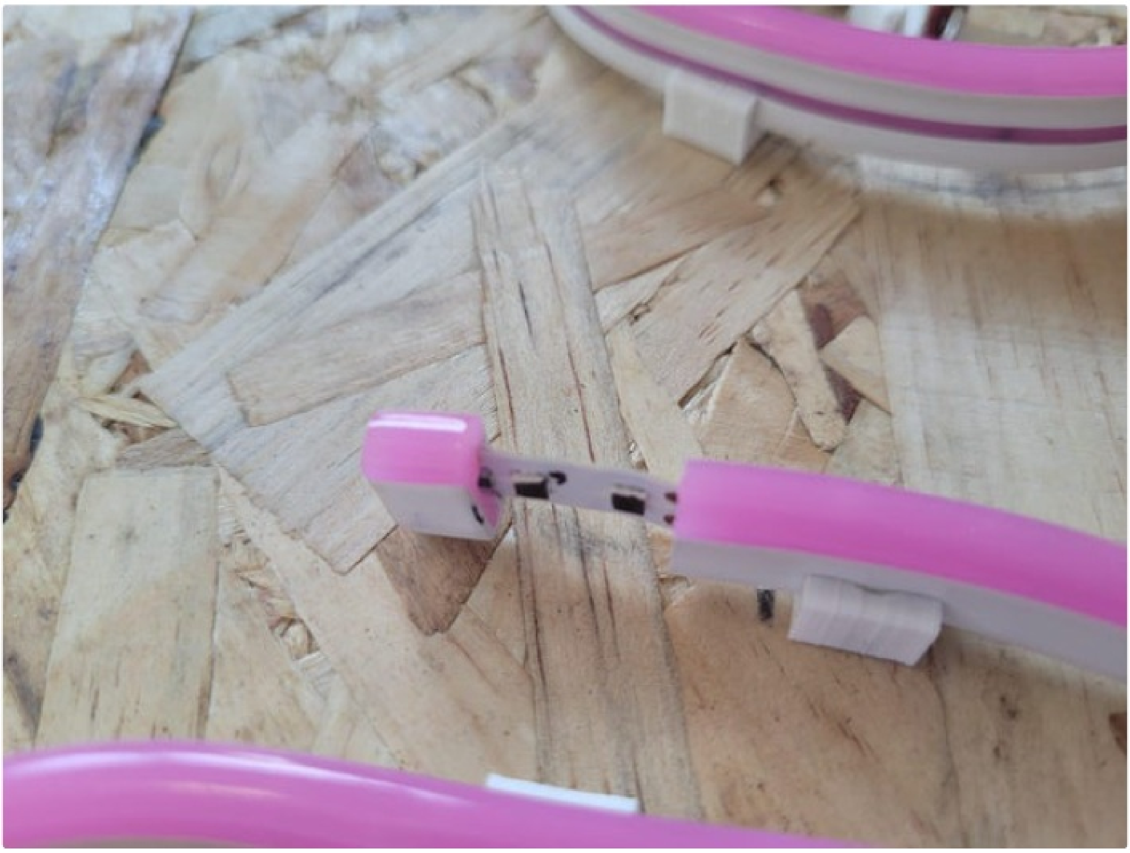


Step 1: Draw Out the Design

To start with, choose the design for the text to display. Select a font that doesn't have too tight curves as it will be difficult to bend the LED strip around. Project the chosen design on the backboard and trace out the lettering with a pencil. Keep stray animals outside the room to speed up the process. If there is no access to a projector, print out the letters on paper and stick them to the board or freehand it. To start off with you need to choose your design for the text you want displayed. You can get all sorts of fonts online but you generally want something that doesn't have too tight curves as it will be diffcult to bend the LED strip around. I found this font to be most suitable for my needs. **https://www.fontspace.com/sunset-club-font-f53575** Once you have chosen a design project it onto your back board, in my case it was a sheet of OSB. Then trace out the lettering with a pencil. Keeping stray animals outside of the room will speed up the process. If you do not have access to a projector you could also print out the letters on paper and stick them to the board or just freehand it.

Step 2: Attach the LED Strips
Next, cut the LED tape into strips for each part of the letters. Cut the tape at specific points for all LEDs to function, usually after every third LED. Design clips to hold onto the strips and attach them to the backboard with small screws. 3D print the clips, or use cable clips or nails to hold the strips in place. For a lower case ‘i,’ cut out a section of silicone around the LEDs and cover a couple of LEDs to create the gap and dot above the body of the letter.

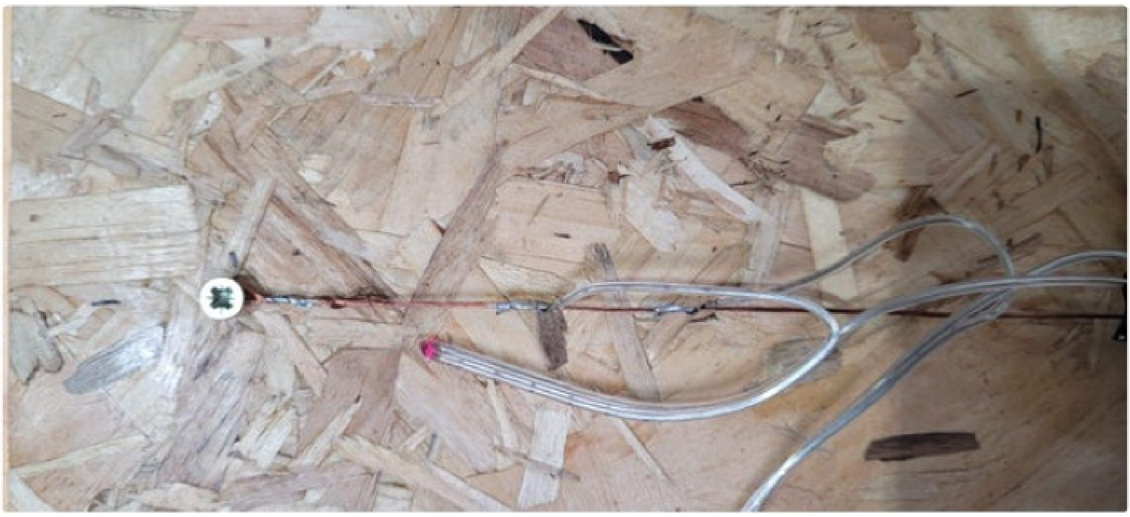
Now you will need to cut the LED tape into strips for each part of the letters. If you have worked with LED tape before you will know that you need to cut the tape at specific points for all LEDs to function, usually after every third LED. This means you may need to make the strips slightly shorter or longer than the section you have just traced out, but with a bit of messing and moving things around you can make the sign look good. I designed some clips on fusion 360 to hold onto the strips and attach them to the back board with some small screws, you can 3D print as many as you need. They are small so fairly quick and easy to print. If you don’t have access to a 3D printer you could just use some cable clips or nails to hold the strips in place. For a lower case ‘i’ you can cut out a section of the silicone around the LEDs and cover a couple of the LEDs to create the gap and dot above the body of the letter.

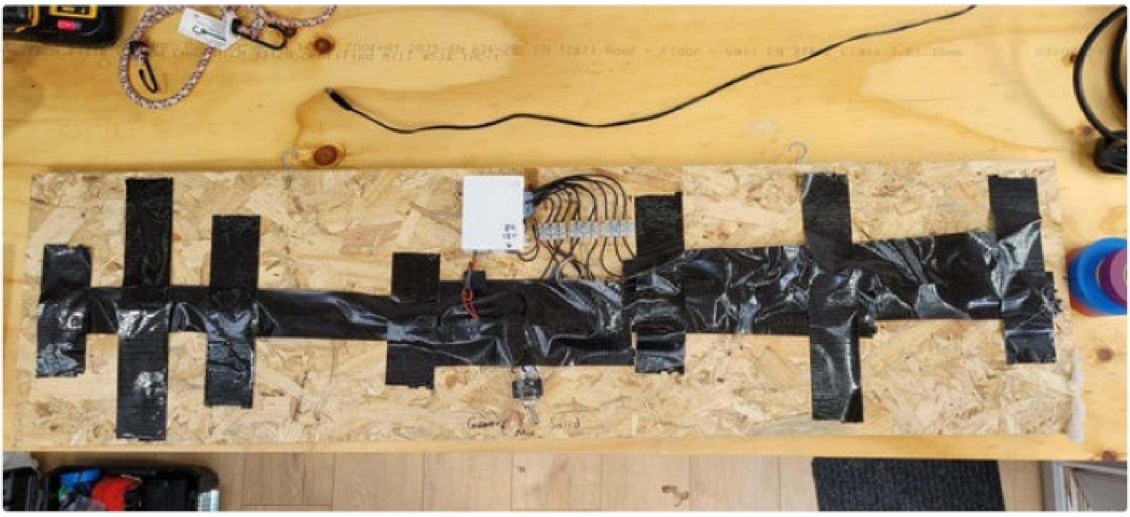


- https://www.instructables.com/FQJ/L28J/LEWU1BWC/FQJL28JLEWU1BWC.stl

Step 3: Wiring Up the LEDs
As the sign can light letters up individually, connect wires from each letter to a single point on the backside of the board. Drill a hole at one end of each section of LED strips and solder a length of double wire to the 12V and GND on each strip. Pass the other end through the small hole. Fix a bare wire along the length of the backside of the board to reduce the amount of cabling required. Connect all of the positive wires to it, making the whole sign much like a common anode 7 segment LED display. Bring over all the common wires and individually connect them to a terminal block. Group together common wires for letters containing more than one segment, such as the letter M. Once all these steps are followed correctly, the Dynamic Neon Arduino Driven Sign is ready to be used as per user requirements.

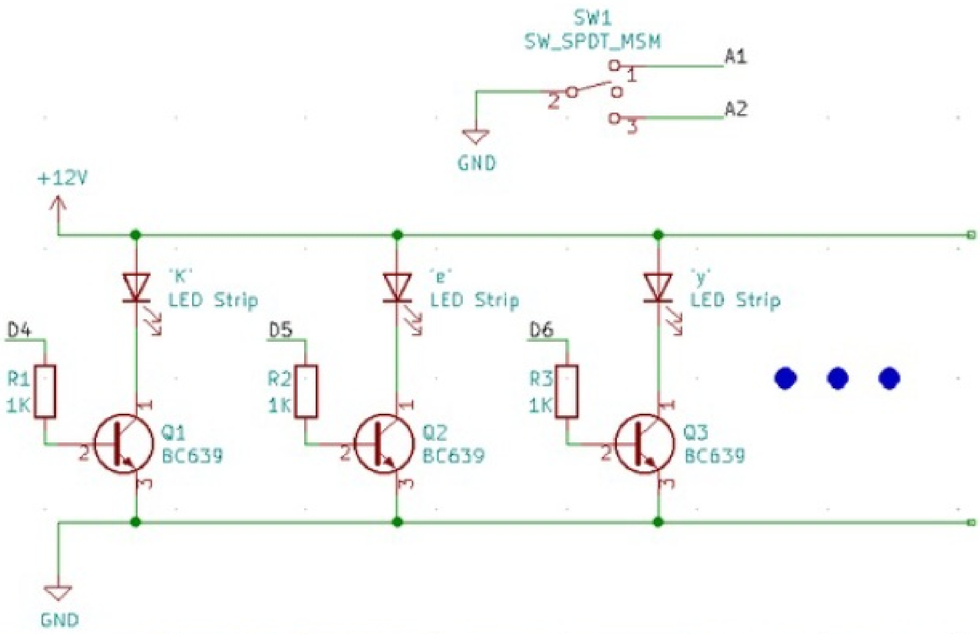
As the sign can light letters up individually you will need to connect wires from each letter to a single point on the back side of the board. At one end of each section of the LED strips, drill a hole just big enough to let the cable through. Solder a length of the double wire to the 12V and GND on each strip and pass the other end thought the small hole. To reduce on the amount of cabling required I fixed a bare wire along the length of the back side of the board and connected all of the positive wires to it, thus making the whole sign much like a common anode 7 segment LED display. All the common wires are then brought over and individually connected to a terminal block. Some letters contain more than one segment such at the letter M, the common wires for this can just be grouped together. All the wires can then be covered in tape to protect them from snagging, and to make it look a bit neater. The back sideof the display does looks a bit crude, but it was made under a tight time schedule and no one will see this anyway except you.

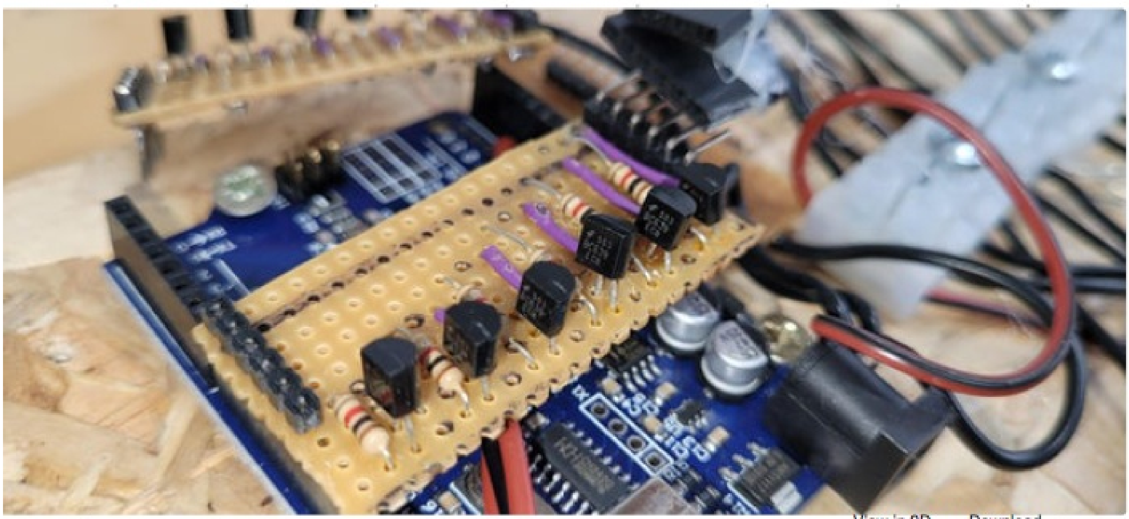




Step 4: Circuitry

An Arduino Uno is used to control each letter, however the GPIO pins on the Arduino cannot sink or source enough current to power the LEDs, so some additional driver circuitry is required. A low side transistor switch can be used to turn the letters on and off. The collector is connected to the low side of each letter, emitter to ground and the base to each GPIO pin of the Arduino via a 1k resistor. Following the circuit diagram you can include as many transistor switches as you have letters on your sign. I made a header board with the transistors to fit neatly on top of the Arduino. If you want more letters than the Uno has GPIO pins available you could upgrade to an Arduino Mega or use an IO expander such as the MCP23017. The 12V cable that goes to all the LED strips is then connected to the back of the positive pin of the barrel connector on the Uno. This way a single 12V DC power supply can be used for the LEDs and Arduino, do make sure the chosen supply can provide enough current for all the LEDs. The last past of the circuitry is to attach a SPDT On-Off-On switch to toggle between the different modes. The common of the switch is connected to GND and the other two pins are connected directly to A1 and A2 and will take advantage of the internal pull up resistors on these pins. I also designed an enclosure that can be 3D printed and attached to the back of the Arduino to provide it with a bit of protection.



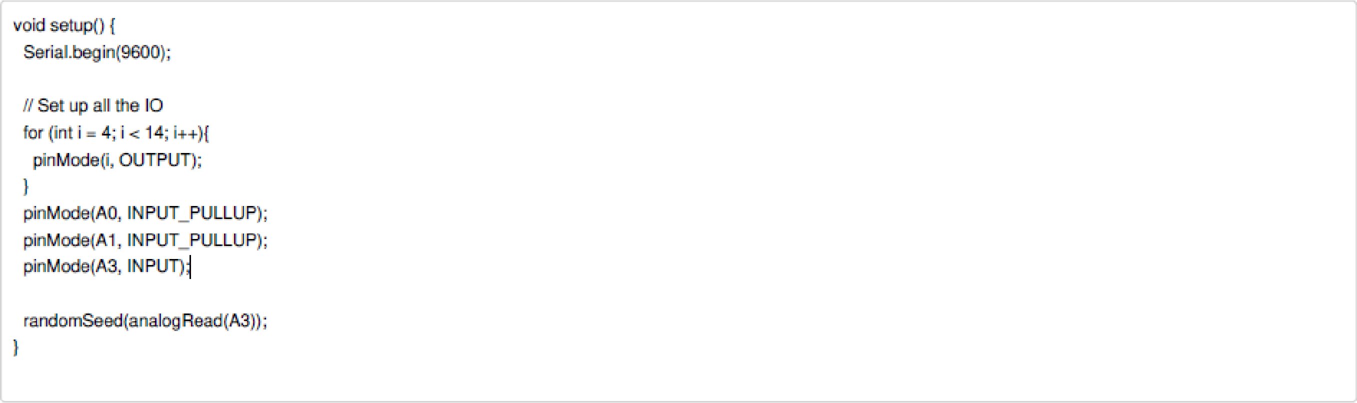


View in 3D Download

https://www.instructables.com/FWN/AK0G/LF14DNFB/FWNAK0GLF14DNFB.stl

Step 5: Software

Now the sign has been constructed and electronics connected, the Arduino can be programmed to produce the groovy patterns. The code is fairly simple, I have written several different functions to light the sign in a variety of ways such as scrolling side to side, flashing words and randomly turn on and off different letters. If you are using different words to my sign you'll need to modify the software slightly so the functions know which IO pins are grouped for each word. For my set up the IO connections to the letters are 4 = 'K', 5 = 'e', 6 = 'y'… The initialisation of the code sets all the digital pins controlling letters to outputs and the two analogue pins connected to the switch as inputs with an internal pullup. A3 is left floating so it can be used as the seed for the random number generation.

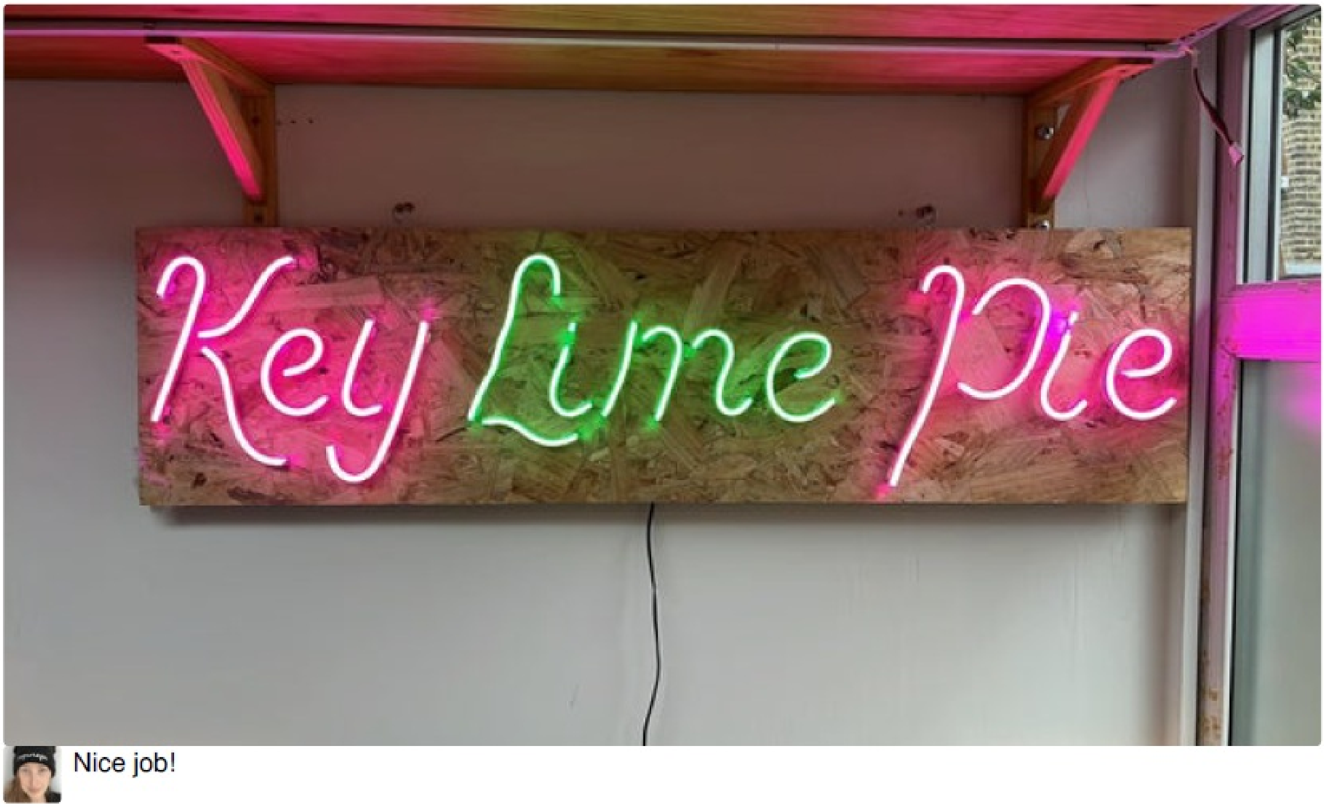


```
void setup() {
  Serial.begin(9600);

  // Set up all the IO
  for (int i = 4; i < 14; i++){
   pinMode(i, OUTPUT);
  }
  pinMode(A0, INPUT_PULLUP);
  pinMode(A1, INPUT_PULLUP);
  pinMode(A3, INPUT);

  randomSeed(analogRead(A3));
}
```

The main loop then reads the status of the switch and will run one of three options depending on its orientation. It will either turn all the LEDs on, cycle through random patterns or alternate between between all on for 60 seconds and patterns for 60 seconds. Again as you are likely to be using different words you'll need to modify the functions that light the individual words, these can be found at the bottom of the code.

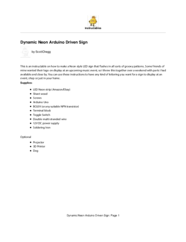
Step 6: All Done!

Finally you should have a great centre piece to put on display in all sorts of locations. Future improvements – based on feed back I have received it would be handy to be able to control the brightness of the sign. This could be done by using a P channel MOSFET switch on the high side of the LEDs and connecting it to one of the PWM pins on the Arduino, varying the duty cycle would then adjust the brightness. If I get round to implementing this I'll update these instructions.



Nice job!

Documents / Resources

	[instructables Dynamic Neon Arduino Driven Sign](#) [pdf] Instructions Dynamic Neon Arduino Driven Sign, Neon Arduino Driven Sign, Arduino Driven Sign, Driven Si gn, Sign

References

- [Yours for the making - Instructables](#)
- [Dynamic Neon Arduino Driven Sign : 6 Steps (with Pictures) - Instructables](#)
- [ScottChegg's Profile - Instructables](#)
- [content.instructables.com/FOJ/4SDB/LF02XZBX/FOJ4SDBLF02XZBX.ino](#)
- [content.instructables.com/FQJ/L28J/LEWU1BWC/FQJL28JLEWU1BWC.stl](#)
- [content.instructables.com/FWN/AK0G/LF14DNFB/FWNAK0GLF14DNFB.stl](#)
- [Sunset Club Font | typefactory | FontSpace](#)