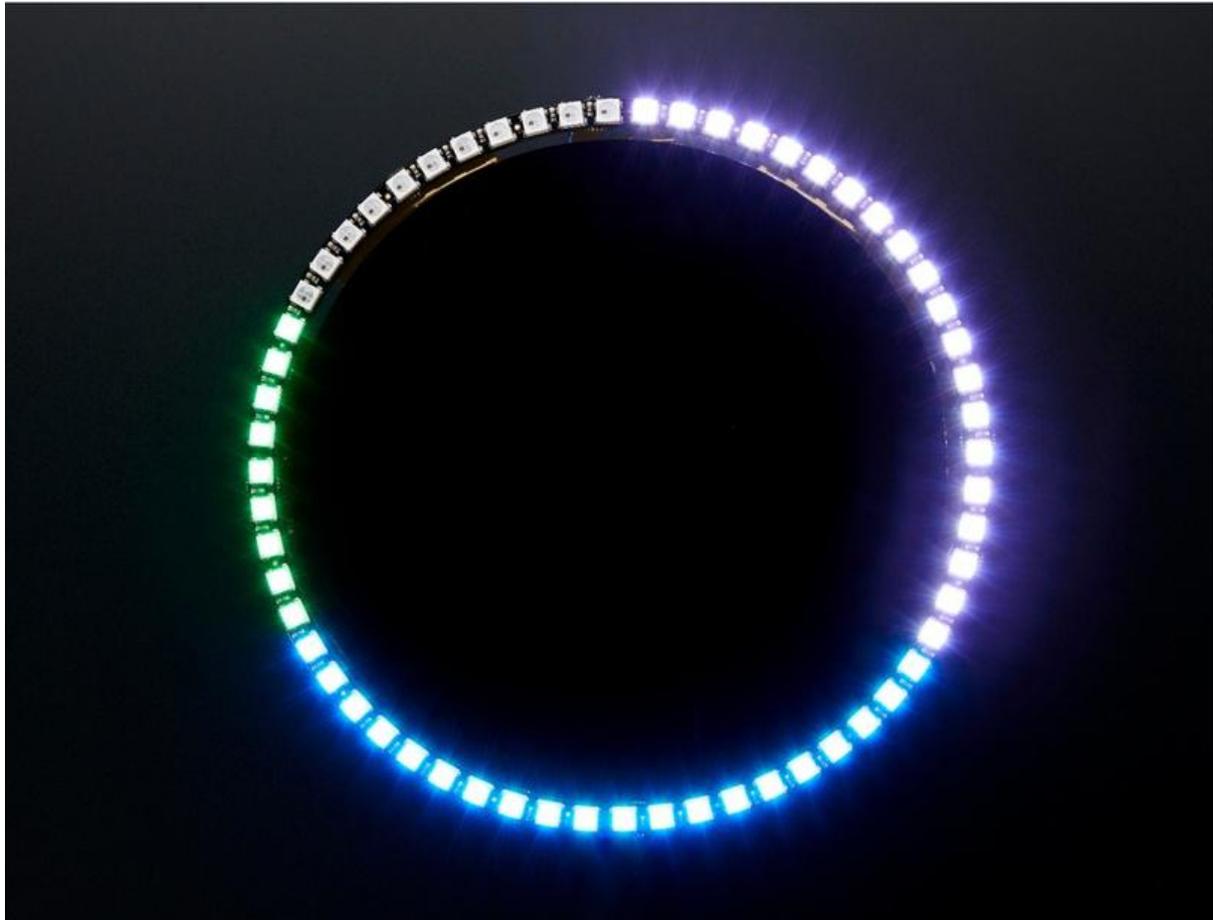




NeoPixel 60 Ring Wall Clock

Created by Andy Doro



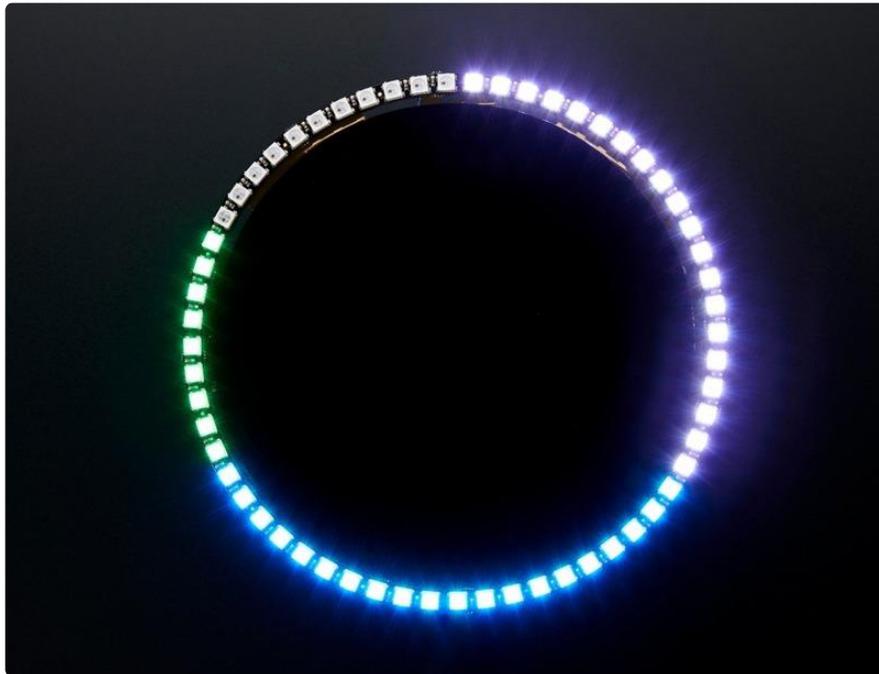
<https://learn.adafruit.com/neopixel-60-ring-clock>

Last updated on 2021-11-15 06:11:27 PM EST

Table of Contents

Overview	3
Building the Circuit	5
Code	7
• Understanding the Code	7
•	7
• Brightness	7
• Start Pixel	8
Finishing It Up	8

Overview



This project uses the [DS1307 Real Time Clock](http://adafru.it/264) (<http://adafru.it/264>) with an Arduino to display the time as a series of colored arcs.

With the new [NeoPixel 60 Ring](http://adafru.it/1768) (<http://adafru.it/1768>), we have a perfect display for presenting the time with RGB LEDs!

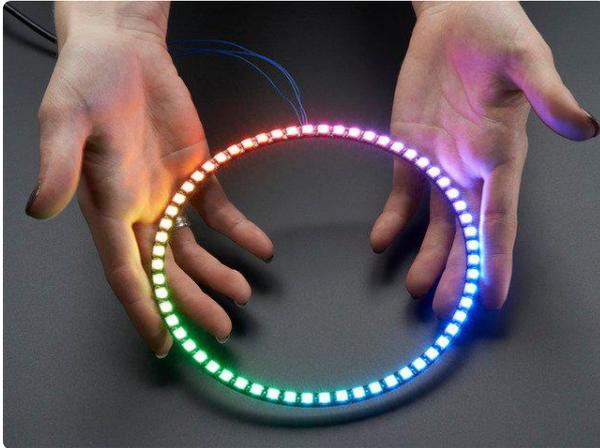
Hours are represented by red, minutes by green, seconds by blue. The arcs overlap and the colors mix. Minutes and seconds are each represented by a single green or blue LED, respectively. In the 12 hour version, a single red LED represent 24 minutes. The image above represents 3:48.

To build this project you will need:

- 4 x [NeoPixel 1/4 60 Ring](http://adafru.it/1768) (<http://adafru.it/1768>) - so you can make a NeoPixel 60 Ring
- [DS1307 Real Time Clock breakout board kit](https://adafru.it/sd0) (<https://adafru.it/sd0>)
- [Arduino Uno](http://adafru.it/50) (<http://adafru.it/50>), [Metro](http://adafru.it/2466) (<http://adafru.it/2466>) or other Arduino compatible microcontroller. This will even work with the [Trinket 5V](http://adafru.it/1501) (<http://adafru.it/1501>)!
- [Adafruit Perma-Proto Half-sized Breadboard PCB](http://adafru.it/1609) (<http://adafru.it/1609>)

If you are new to microcontrollers and Arduino, I would recommend starting out by building the circuit using the [Arduino Uno](http://adafru.it/50) (<http://adafru.it/50>) and a [breadboard](http://adafru.it/239) (<http://adafru.it/239>).

If you are more familiar with Arduino, you can use a [Trinket 5V](http://adafru.it/1501) (<http://adafru.it/1501>) and perfboard to build a more permanent electronics project.



[NeoPixel 1/4 60 Ring - 5050 RGB LED w/ Integrated Drivers](https://www.adafruit.com/product/1768)

The biggest NeoPixel Ring yet! With four of these you can make a huge ring with 60 ultra bright smart LED NeoPixels are arranged in a circle with a 6.2" diameter. Each...

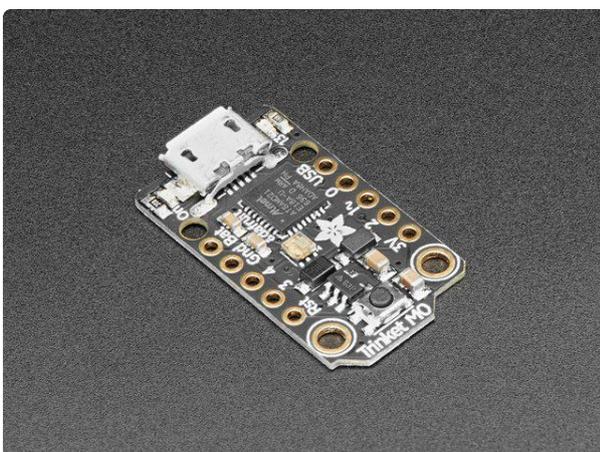
<https://www.adafruit.com/product/1768>



[Adafruit DS1307 Real Time Clock Assembled Breakout Board](https://www.adafruit.com/product/3296)

This is a great battery-backed real time clock (RTC) that allows your microcontroller project to keep track of time even if it is reprogrammed, or if the power is lost. Perfect for...

<https://www.adafruit.com/product/3296>

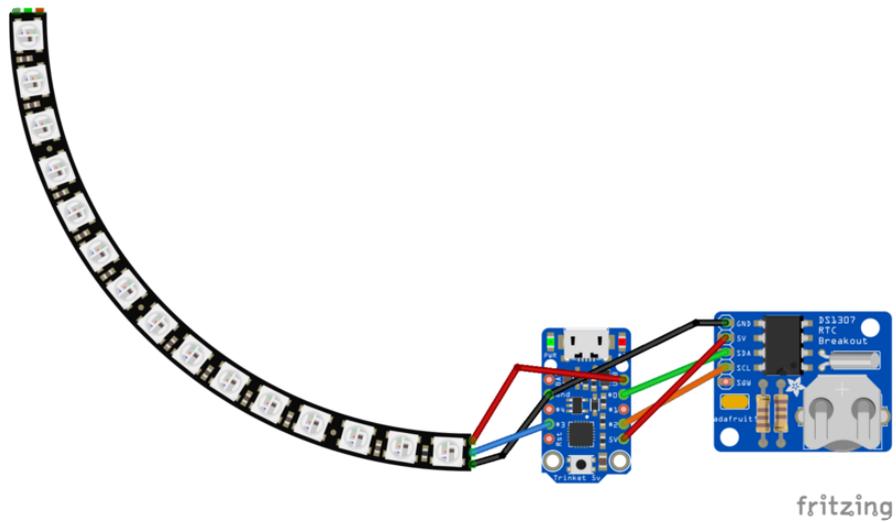


[Adafruit Trinket M0 - for use with CircuitPython & Arduino IDE](https://www.adafruit.com/product/3500)

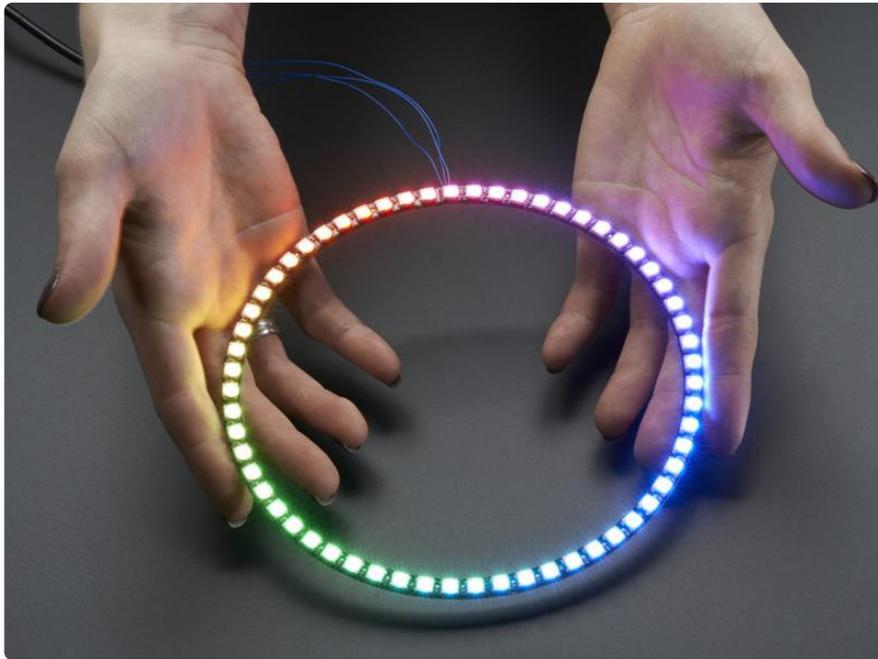
The Adafruit Trinket M0 may be small, but do not be fooled by its size! It's a tiny microcontroller board, built around the Atmel ATSAM21, a little chip with a lot...

<https://www.adafruit.com/product/3500>

Building the Circuit



Solder 4 pieces of the the NeoPixel 1/4 60 Ring together, connecting GND to GND, 5V to 5V and DIN to DOUT. Butt the ends together and use a small piece of stripped wire as a 'bridge' to give some mechanical support.



Leave one pair of DIN and DOUT disconnected- this will be where the NeoPixel strand begins and ends. The microcontroller data pin will connect to the DIN here.

The ring is very delicate! So please mount onto a backing or use care when moving around!

NeoPixel Ring connects as follows:

- +5V → 5V pin on Arduino or USB pin on Trinket (red wire)
- GND → GND pin on Arduino or Trinket (black wire)
- DIN → D3 pin on Arduino or #3 Trinket (blue wire)

DS1307 RTC connects as follows:

- +5V → 5V on Arduino or Trinket (red wire)
- GND → GND on Arduino or Trinket (black wire)
- SDA → Analog 4 on Arduino or #0 on Trinket (green wire)
- SCL → Analog 5 on Arduino or #2 on Trinket (orange wire)

Assemble the DS1307 breakout following [these instructions \(https://adafru.it/ixe\)](https://adafru.it/ixe) if it's not pre-assembled.

Code

Important note: when running this code, you should probably use an external power supply (say 9VDC into the Arduino). When I ran the code on USB power, the NeoPixels drew too much current and this caused the RTC to provide strange numbers to the Arduino.

[Download the code \(https://adafru.it/fFC\)](https://adafru.it/fFC) and uncompress in your Arduino sketch folder.

This Arduino sketch relies on a few Arduino libraries:

- [RTCLib \(https://adafru.it/c7r\)](https://adafru.it/c7r)
- [Adafruit_NeoPixel \(https://adafru.it/aZU\)](https://adafru.it/aZU)
- [DST_RTC \(https://adafru.it/BR3\)](https://adafru.it/BR3)

Follow the [DS1307 RTC breakout tutorial \(https://adafru.it/dhu\)](https://adafru.it/dhu) to learn how to install the RTC library and set the time on the DS1307 RTC.

You will also need to install the [NeoPixel library \(https://adafru.it/dhv\)](https://adafru.it/dhv).

The [DST_RTC library \(https://adafru.it/BR3\)](https://adafru.it/BR3) is used to keep the time accurate for Daylight Saving Time.

See the following tutorial to [learn how to install Arduino libraries \(https://adafru.it/dit\)](https://adafru.it/dit).

Understanding the Code

Brightness

The clock operates with two different brightness levels, so that it can switch to a dimmer level at nighttime. The two brightness levels are `DAYBRIGHTNESS` and `NIGHT BRIGHTNESS`. These values can be 0-255 although with 0 you won't be able to see anything.

The daytime / nighttime cutoffs are set with the `MORNINGCUTOFF` and `NIGHTCUTOFF` values, which define which hours the brightness transitions take place.

```
// nighttime dimming constants
// brightness based on time of day- could try warmer colors at night?
#define DAYBRIGHTNESS 64
#define NIGHTBRIGHTNESS 20

// cutoff times for day / night brightness. feel free to modify.
#define MORNINGCUTOFF 7 // when does daybrightness begin? 7am
#define NIGHTCUTOFF 22 // when does nightbrightness begin? 10pm
```

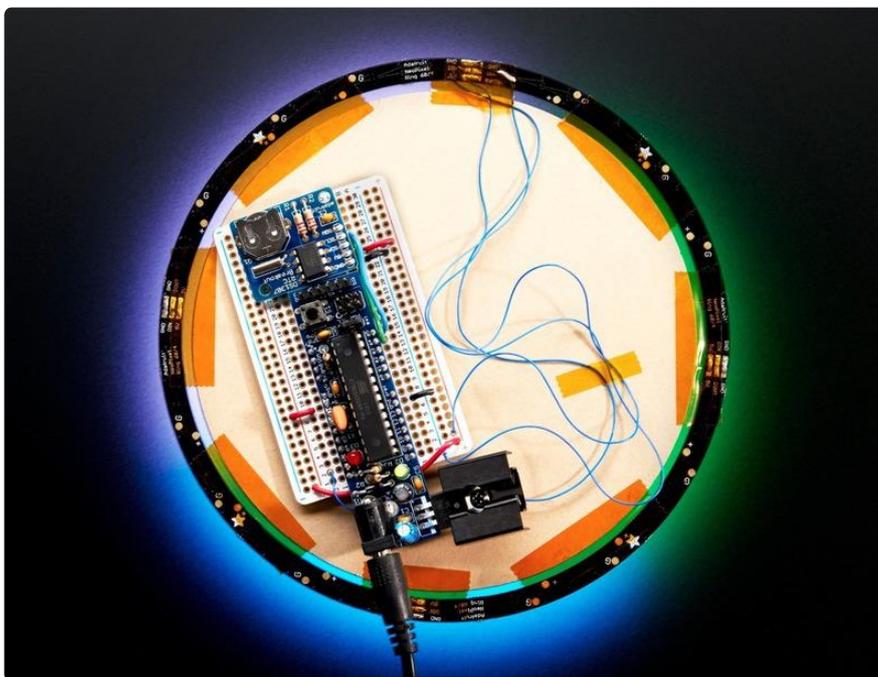
Start Pixel

You can define STARTPIXEL to define where the "midnight" point on the clock is. This can be useful, for example, if you have the clock in [an enclosure](https://adafru.it/xXD) (<https://adafru.it/xXD>) and you're unable to physically rotate the NeoPixel Ring.

```
#define STARTPIXEL 0 // where do we start on the loop? use this to shift the arcs
if the wiring does not start at the "12" point
```

A value of 0 would mean the colored arcs would begin and end where the DIN wire connects from the microcontroller to the NeoPixel Ring. You can change the value from 0-59 to change where the arcs begin.

Finishing It Up



This is the back of the clock, pictured here with the discontinued USB Boarduino. As stated before, this project works great with the [Trinket 5V](http://adafru.it/1501) (<http://adafru.it/1501>), [Arduino Uno](http://adafru.it/50) (<http://adafru.it/50>), [Metro](http://adafru.it/2466) (<http://adafru.it/2466>) or other Arduino compatible microcontroller board!

I used a spare circular piece of black plexi to hide the electronics behind, since I'm just hanging the clock near a desk.

If you want to you can create a more sophisticated enclosure for this clock. [arradan](https://adafru.it/Cd2) (<https://adafru.it/Cd2>) created [this enclosure to 3D print](https://adafru.it/xXD) (<https://adafru.it/xXD>) on Thingiverse!



