Animated GIF Player for Matrix Portal

Created by Phillip Burgess

https://learn.adafruit.com/animated-gif-player-for-matrix-portal

Last updated on 2022-12-01 03:58:13 PM EST

Table of Contents

Overview

There's something eye-catching about LED matrix graphics that you don't get with a normal LCD display. Maybe it's the brightness or the ultra-saturated colors. Perhaps that's why they're increasingly popular in advertising signage.

This guide uses an Adafruit MatrixPortal M4 and RGB LED matrix to create an animated GIF player that's drag-and-drop easy, like a pixelated version of a digital photo frame. Use it to make a holiday window display or to advertise your favorite shawarma cart.

This is a different approach than used in our Sprite Sheet Animation () and Tombstone () guides. There, the animation is controlled by CircuitPython code…letting you control speed or interactivity or anything else you can dream up. But when you'd like something simpler, animated GIFs are easy and ubiquitous…you can likely find some ready-made with a few internet searches. Toss a bunch on the Matrix Portal flash drive and it plays them back in a loop.
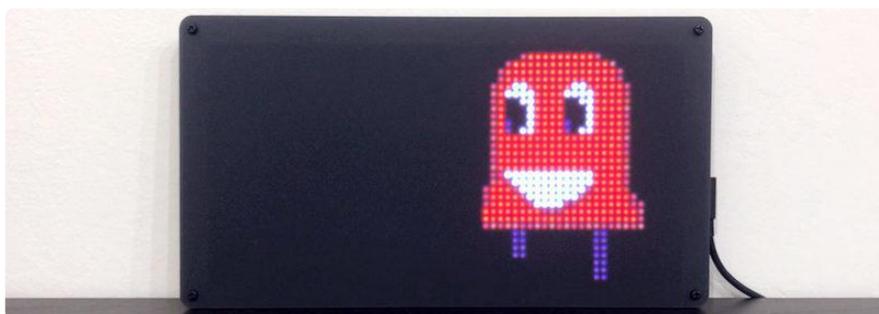
Quickstart

The GIF player loads images from the Matrix Portal's flash filesystem. Although it's an Arduino-based project, it's necessary to temporarily install CircuitPython on the board to set up that drive for the first time. If you've previously installed CircuitPython, then these steps are already done...

1. Visit the CircuitPython download page for Matrix Portal M4 () and download a current .UF2 file.
2. Connect the MatrixPortal board to your computer via USB cable.
3. Double-click the board's reset button.
4. After a few seconds, the onboard NeoPixel should turn green and a drive called MATRIXBOOT will appear on your computer.
5. Drag the CircuitPython .UF2 file to this drive and wait several seconds for it to install. The MATRIXBOOT drive will be replaced by a new drive called CIRCUITPY.

Create a folder called gifs on this drive. That's where we'll put images for playback.

NOTE: you must create the gifs folder on the CIRCUITPY drive and put images in there, or nothing will happen.

Then download the precompiled .UF2 file for this project:

MATRIX_PORTAL_ANIMATED_GIF.UF2

NOTE: this ready-made version works with a 64x32 pixel matrix. The code can handle other sizes but you'll need to adjust and compile the code yourself in that case, explained on the "Compiling and Customizing" page.

Installing this is similar to CircuitPython above:

1. Connect the MatrixPortal board to your computer via USB cable.

2. Double-click the board's reset button.
3. After a few seconds, the onboard NeoPixel should turn green and a drive called MATRIXBOOT will appear on your computer.
4. Drag the GIF player .UF2 file to this drive and wait several seconds for it to install. The MATRIXBOOT drive will be replaced by a new drive called CIRCUITPY.

Now you just need to load up the gifs folder with suitable images, and tap the reset button to restart.

CAUTION: copying GIFs to the CIRCUITPY drive while the GIF player is running sometimes causes problems, occasionally even clearing the drive. For added safety, temporarily install the Circuit Python .UF2, set up the gifs folder on the drive, then load up the GIF player .UF2. Always keep backups of your images and code!


Prepare GIFs
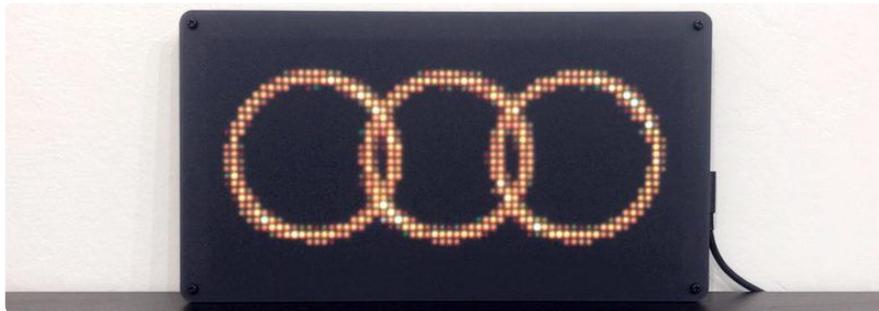
The GIF player does not scale images. If they're a little bit smaller or larger than the LED matrix, they'll be centered or cropped to fit. Some GIFs do fine with a little cropping…but if they're much larger than the matrix, as most GIFs you'll find on the internet are, they'll still play but won't make any sense with such a small area clipped out. You may have to do some prep work…

These techniques have already been covered in other guides. Rather than repeat ourselves, we'll link to those guides and provide some commentary with how things may be different when using an LED matrix.

A similar project for the original PyPortal (like a MatrixPortal with an LCD display) explains the use of an online tool — EZGIF — for resizing and optimizing GIF images, so you don't need to own any special image editing software () (opens in new window).

Use 64 and 32 for the width and height settings (rather than 320 and 240) if that's what matches your matrix. Keep the other settings as recommended in that guide. With the smaller pixel dimensions of the LED matrix, the Optimize and Reduce Frames steps can be ignored. It's no trouble decoding these small GIFs at full speed!

GIFs played on the LED matrix may appear overly bright and washed-out. A different guide explains how to adjust for this () (opens in new window). Rather than an online tool, this guide uses software installed on your computer to make the adjustment. This approach does require some command-line chops...but...since this can handle the same scaling and cropping options as the EZGIF method above, you don't need to run your image through multiple tools, it's a quicker one-stop solution.
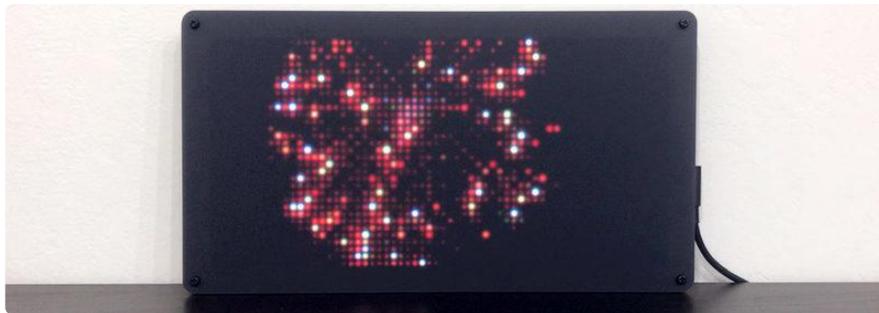
IMPORTANT NOTE: the AnimatedGIF library does not handle interlaced GIFs (this is a tradeoff for handling arbitrarily long/complex animations). When saving or converting GIF images, be certain that interlacing is OFF.

If using ImageMagick () for conversion, this can be done with the -interlace None option, as shown in the matrix guide linked above.

Compiling and Customizing

If the GIF player doesn't quite do what you want (different matrix size, etc.), the source code can be tweaked and recompiled.

This assumes you already have the Arduino development environment installed and Matrix Portal M4 board support added. If you've not done this before, these steps are described in the MatrixPortal M4 introductory guide () (opens in new window).

Additionally, this requires that the Adafruit_Protomatter and AnimatedGIF Arduino libraries be installed. This process is also explained in the guide linked above, or in the Adafruit_Protomatter guide () (opens in new window).

Once installed in the Arduino IDE, you can find the animated GIF player among the Protomatter library examples…

File→Examples→Adafruit Protomatter→animated_gif

You can also find the project source code on Github (), but will still need the Protomatter library installed.

Changing the GIF folder and playback time

Near the top of the code, starting around line 26, you'll see these two lines:

```
char GIFpath[] = "/gifs";      // Absolute path to GIFs on CIRCUITPY drive
uint16_t GIFminimumTime = 10; // Min. repeat time (seconds) until next GIF
```

The first of these is the folder name containing the GIF images to play. By default this is `gifs`, but you can change that here if you like.

Second line tells how long to repeat each GIF. This is a minimum time, in seconds. With the default value of `10` here, a short 1-second GIF would be repeated 10 times before advancing to the next image…while a long 10+ second GIF will play only once. GIF playback is not truncated to this time, they will always play back in their entirety at least once.

Using different matrix sizes

The next couple of lines define the matrix dimensions…

```
#define WIDTH  64 // Matrix width in pixels
#define HEIGHT 32 // Matrix height in pixels
// Maximim matrix height is 32px on most boards, 64 on MatrixPortal if the
// 'E' jumper is set.
```

NOTE: 64x64 matrices require soldering between two pads on the MatrixPortal board. This is explained in the MatrixPortal guide () (see "Address E Line Jumper" at the bottom). Other matrix sizes (64x32 and 32x16) are plug-and-play with just the code change.

Make sure you have the correct board type selected:

Tools➜Board➜Adafruit Matrix Portal M4

Compile and upload to the board, and it should work at the new size or new folder & time settings.

HELPFUL TIP: remember how you double-tap reset to install a .UF2 file onto the MatrixPortal board? This works both ways! Once you've got the code configured and running how you like, double-tap the reset button and copy the file CURRENT.UF2 off the board to your computer. Now it's easy to reinstall your custom version any time. This file can be renamed something descriptive if you like, just keep the .UF2 at the end.

Example GIFs

These images will look a bit dark in your browser but will play correctly on the LED matrix. Right-click and "Save Image…" to download each GIF. Remember to put these inside the gifs folder on the CIRCUITPY drive.