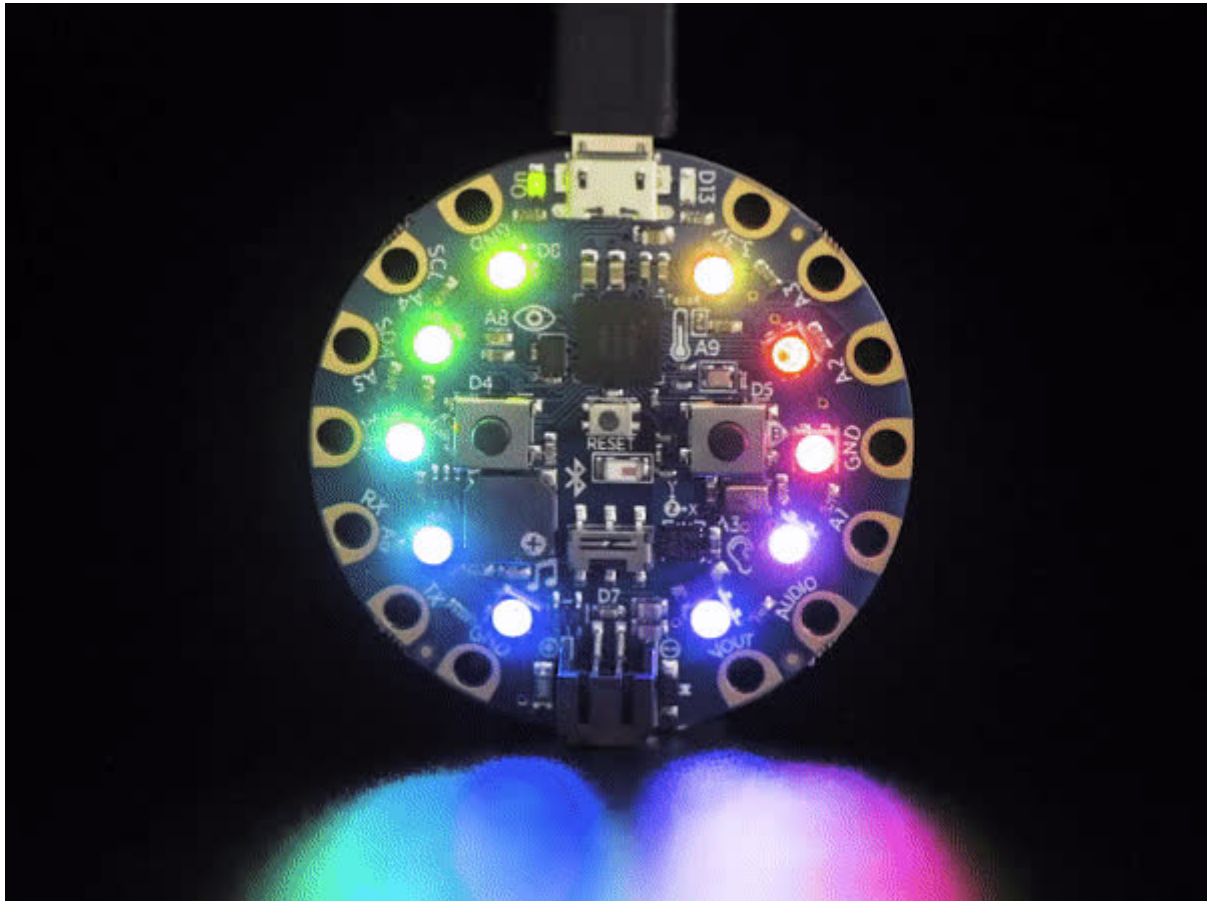




PyLeap device enabled - In Rainbows

Created by phillip torrone



<https://learn.adafruit.com/pyleap-device-enabled-in-rainbows>

Last updated on 2025-02-25 12:02:00 PM EST

Table of Contents

Glide on over to some Rainbows	3
Blink	3
LED Glasses	4

Glide on over to some Rainbows

PyLeap will list the device enabled guides, including this one. Our first stop is using Glider (wireless file transfer) inside of PyLeap to work with BundleFly on the Adafruit Learning System to bundle up and send the files on over! The files include **code.py** and the libraries. For this proof-of-concept we're going to toss a rainbow on over to a Circuit Playground Bluefruit Express.

```
# SPDX-FileCopyrightText: 2021 Phillip Torrone for Adafruit Industries
#
# SPDX-License-Identifier: MIT

import time
import board
from rainbowio import colorwheel
import neopixel

pixels = neopixel.NeoPixel(board.NEOPIXEL, 10, brightness=0.2, auto_write=False)
rainbow_cycle_demo = 1

def rainbow_cycle(wait):
    for j in range(255):
        for i in range(10):
            rc_index = (i * 256 // 10) + j * 5
            pixels[i] = colorwheel(rc_index & 255)
            pixels.show()
            time.sleep(wait)

while True:
    if rainbow_cycle_demo:
        rainbow_cycle(0.05)
```

Blink

This is Blink demo code for PyLeap.

```
# SPDX-FileCopyrightText: 2021 TrevKnows for Adafruit Industries
#
# SPDX-License-Identifier: MIT

import time
import board
import neopixel

pixels = neopixel.NeoPixel(board.NEOPIXEL, 10, brightness=0.2, auto_write=False)
PURPLE = (10, 0, 25)
PINK = (25, 0, 10)
OFF = (0,0,0)

while True:
    pixels.fill(PURPLE)
    pixels.show()
    time.sleep(0.5)
    pixels.fill(OFF)
    pixels.show()
```

```
time.sleep(0.5)
pixels.fill(PINK)
pixels.show()
time.sleep(0.5)
pixels.fill(OFF)
pixels.show()
time.sleep(0.5)
```

LED Glasses

Swirl Demo

```
# SPDX-FileCopyrightText: 2021 ladyada for Adafruit Industries
# SPDX-License-Identifier: MIT

import board
from rainbowio import colorwheel
from adafruit_is31fl3741.adafruit_ledglasses import LED_Glasses
import adafruit_is31fl3741

i2c = board.I2C() # uses board.SCL and board.SDA
# i2c = board.STEMMA_I2C() # For using the built-in STEMMA QT connector on a
microcontroller
glasses = LED_Glasses(i2c, allocate=adafruit_is31fl3741.MUST_BUFFER)

wheeloffset = 0
while True:
    for i in range(24):
        hue = colorwheel(i * 256 // 24 + wheeloffset)
        glasses.right_ring[i] = hue
        glasses.left_ring[23 - i] = hue
    glasses.show()
    wheeloffset += 10
```

Sparkle Demo

```
# SPDX-FileCopyrightText: 2021 Rose Hooper
# SPDX-License-Identifier: MIT

import board
from adafruit_led_animation.animation.sparkle import Sparkle
from adafruit_led_animation.color import PURPLE
from adafruit_led_animation.sequence import AnimationSequence

from adafruit_is31fl3741.adafruit_ledglasses import MUST_BUFFER, LED_Glasses
from adafruit_is31fl3741.led_glasses_animation import LED_Glasses_Animation

i2c = board.I2C() # uses board.SCL and board.SDA
# i2c = board.STEMMA_I2C() # For using the built-in STEMMA QT connector on a
microcontroller
glasses = LED_Glasses(i2c, allocate=MUST_BUFFER)
glasses.set_led_scaling(255)
glasses.global_current = 0xFE
glasses.enable = True

pixels = LED_Glasses_Animation(glasses)

anim2 = Sparkle(pixels, 0.05, PURPLE)

group = AnimationSequence(
    anim2, advance_interval=5, auto_reset=True, auto_clear=True
```

```
)  
while True:  
    group.animate()
```