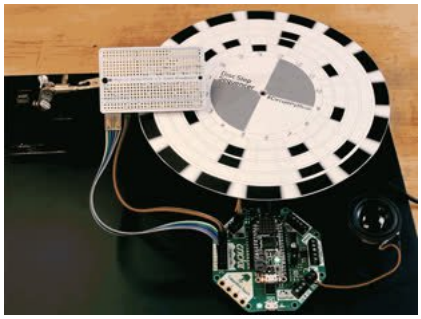
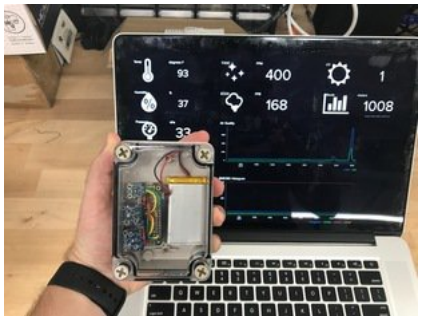
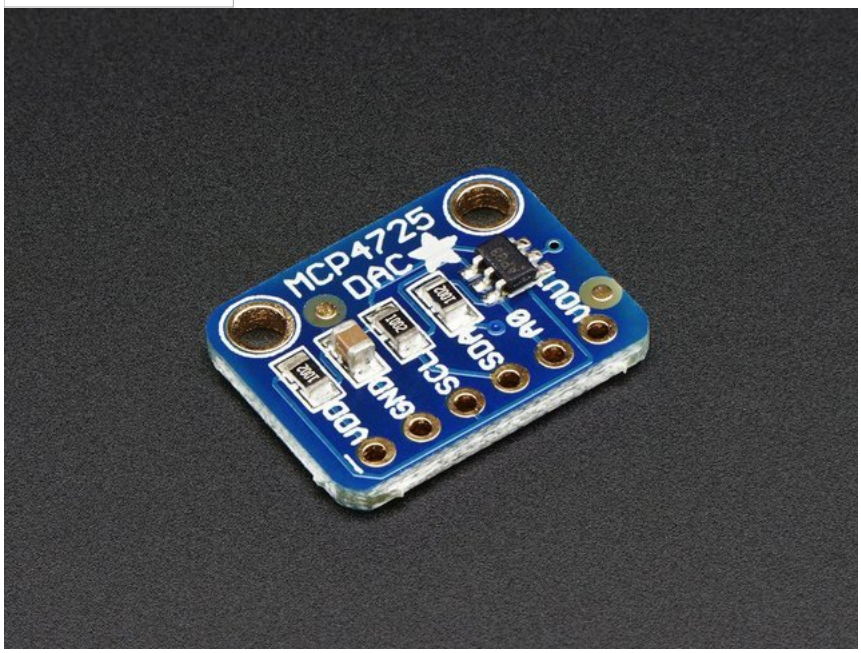
- DIY ADAPTIVE GAME CONTROLLERS
- DANCE-REACTIVE TUTU SPARKLE SKIRT
-  SPINNING DISC STEP SEQUENCER

FEATURED GUIDES
(SEE ALL 26)

- GLOWING SLIME LUNCHBOX
- THE SCREAM: INTERACTIVE SCREAMING PAINTING
-  ADAFRUIT IO ENVIRONMENTAL MONITOR FOR FEATHER OR RASPBERRY PI

- AUTOMATIC CAT TREAT DISPENSER



MCP4725 12-Bit DAC Tutorial

Add an analog output to your microcontroller

- Overview
- Wiring
- Arduino Code
- Python & CircuitPython
  - Python Docs
- Download
-
- Featured Products
- Single Page
- Download PDF

Contributors

lady ada
Tony DiCola
Kattni Rembor

Feedback? Corrections?

ADAFRUIT PRODUCTS MICROCONTROLLERS BREAKOUT BOARDS / AMPLIFIERS/SOUND

Arduino Code

by lady ada

Next up, download the Adafruit MCP4725 library. This library does all of the interfacing, so you can just "set and forget" the DAC output. It also has some examples to get you started

The library is available on GitHub. You can download it by clicking the button below.

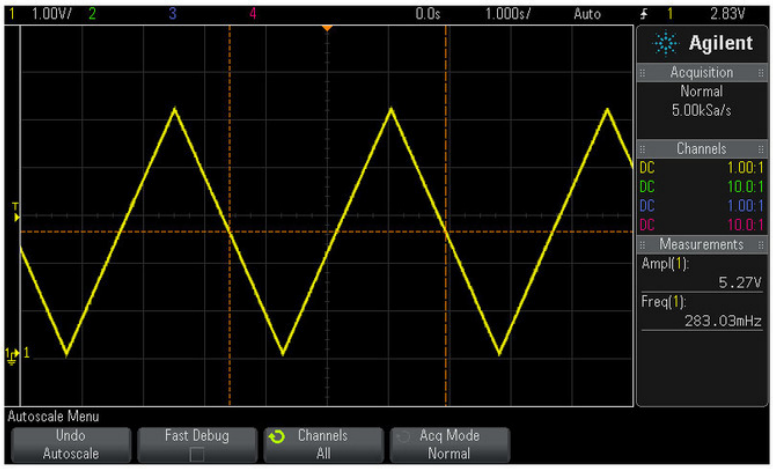
Download Adafruit_MCP4725 Library

Rename the uncompressed folder **Adafruit_MCP4725.** Check that the **Adafruit_MCP4725** folder contains **Adafruit_MCP4725.cpp** and **Adafruit_MCP4725.h**

Place the **Adafruit_MCP4725** library folder your ***sketchbookfolder*/libraries/** folder. You may need to create the **libraries** subfolder if its your first library. You can figure out your ***sketchbookfolder*** by opening up the Preferences tab in the Arduino IDE.

Restart the IDE.

Open up the **File→Examples→Adafruit_MCP4725→trianglewave** sketch and upload it to the Arduino. Then connect your oscilloscope (or an LED + resistor if you don't have access to an oscilloscope)



We also have a sine wave version showing how to use a lookup table to create a more complex waveform.

Using the library

The library is very simple, so you can adapt it very quickly.

First, be sure to call **begin(addr)** where **addr** is the i2c address (default is 0x62, if A0 is connected to VCC its 0x63). Then call **setVoltage(value, storeflag)** to set the DAC output. **value** should range from 0 to 0x0FFF. **storeflag** indicates to the DAC whether it should store the value in EEPROM so that next time it starts, it'll have that same value output. You shouldn't set the flag to true unless you require it as it will take longer to do, and you could wear out the EEPROM if you write it over 20,000 times.

Increasing the speed

One thing thats a little annoying about the Arduino Wire library in this case is it is set for 100KHz transfer speed. In the MCP4725 library we update the speed to 400KHz by setting the TWBR

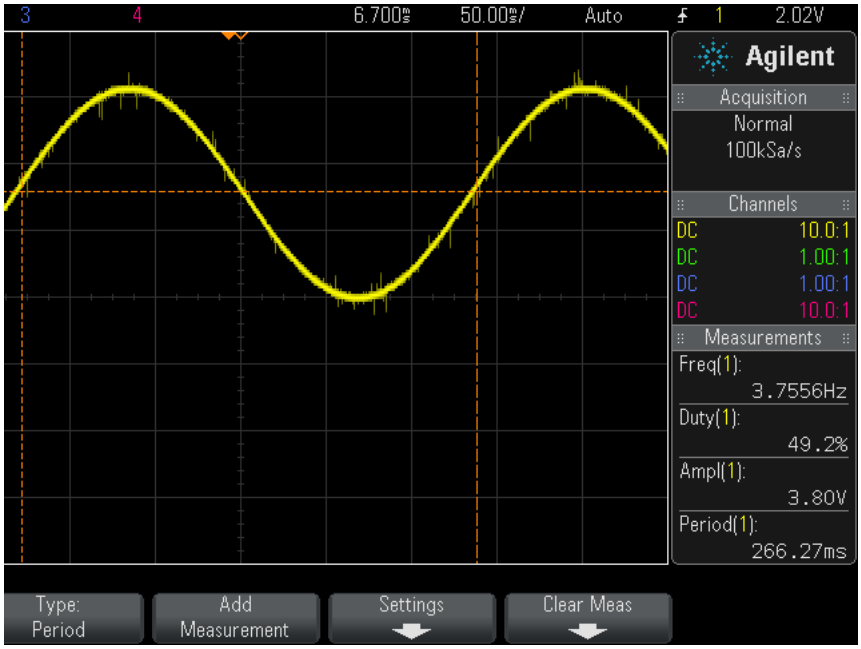
```
TWBR = 12; // 400 khz
```

You can speed this up a bit more, if you'd like, check the ATmega328 datasheet for how to calculate the **TWBR** register.

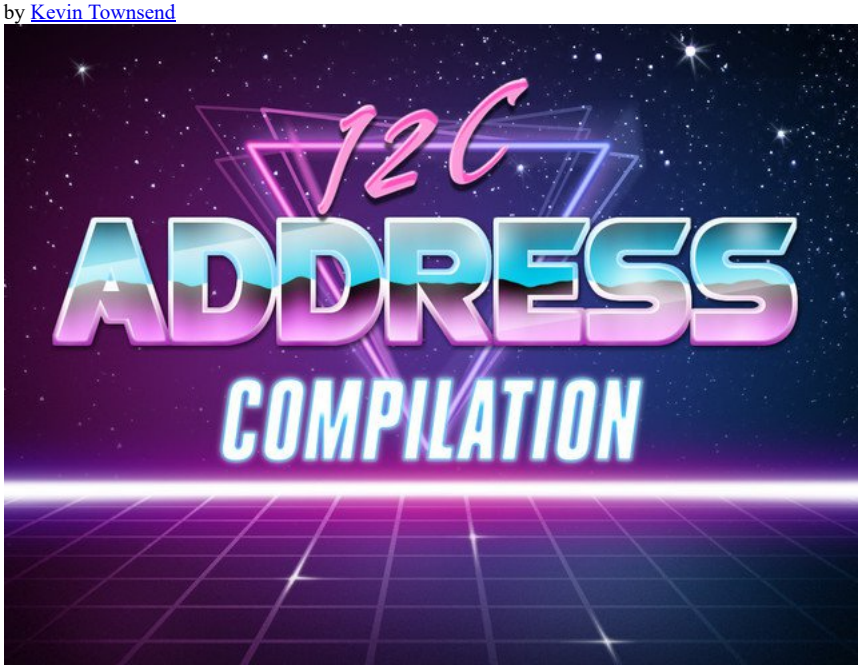
WIRING PYTHON & CIRCUITPYTHON
This guide was first published on Sep 05, 2012. It was last updated on Sep 05, 2012. This page (Arduino Code) was last updated on Jan 27, 2018.

MCP4725 Breakout Board - 12-Bit DAC w/I2C Interface
$4.95 Add to Cart

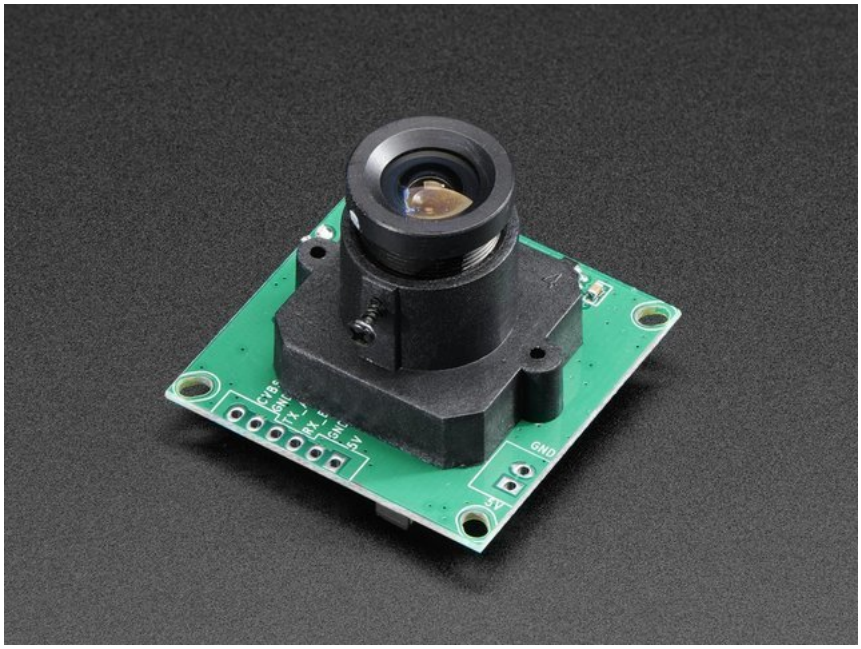
Related Guides



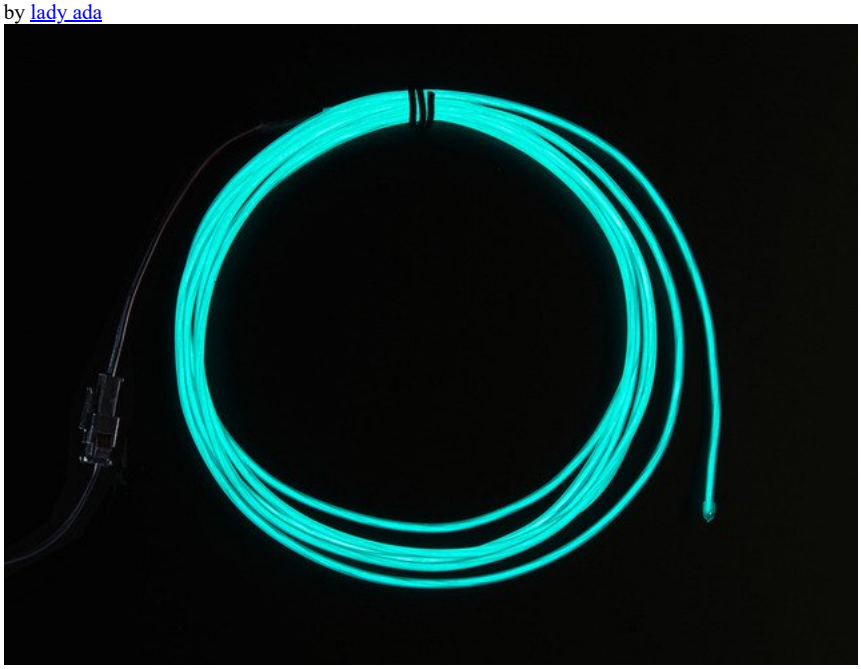
Add a DAC (digital to analog convertor) to your Raspberry Pi
MCP4725 12-Bit DAC with Raspberry Pi
by Kevin Townsend



I2C addresses from 0x00 to 0x7F (inclusive)
I2C addresses!
by lady ada



Snap, Snap!
TTL Serial Camera
by lady ada



Working with Electroluminescent Wire
EL Wire
by lady ada
×

OUT OF STOCK NOTIFICATION

YOUR NAME
YOUR EMAIL
NOTIFY ME

VISA
MasterCard
AMERICAN EXPRESS
PayPal
amazonpayments
DISCOVER