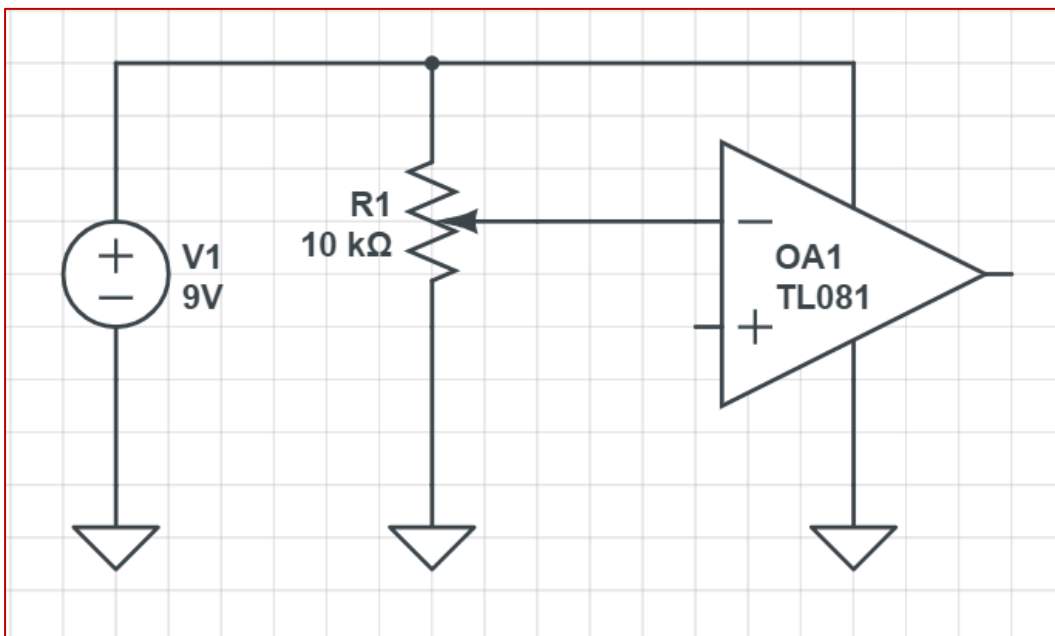


## 7Semi MCP4725 12bit DAC user guide

### Where this breakout board can be used?

The 7semi MCP4725 board is a breakout module for the MCP4725 Digital-to-Analog Converter (DAC), which allows you to convert digital signals into analog voltages. This is useful for controlling analog devices like LEDs, motors, and audio outputs using digital controllers such as Arduino, Raspberry Pi, or other microcontrollers.

*Let's look for one basic example for its implementation in the circuitry.*



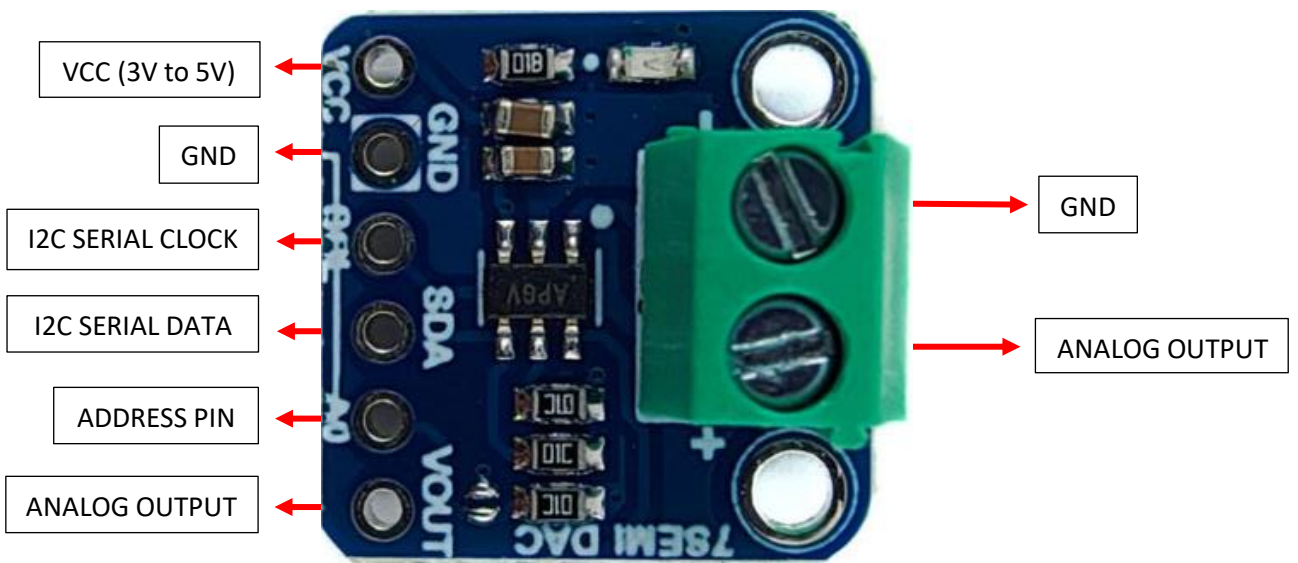
In the above image if we rotate the pot, the reference voltage (analog voltage) given to the opamp's inverting pin will change and corresponding to it the output will vary as per the working of the circuit (it's just a basic example and not included the other components in the circuit).

So, if you use **MCP4725** DAC you don't have to rotate the pot to change the reference voltage, you can use a button interface or change the reference over Wi-Fi if your project has Wi-Fi options. Likewise, there are many applications further.

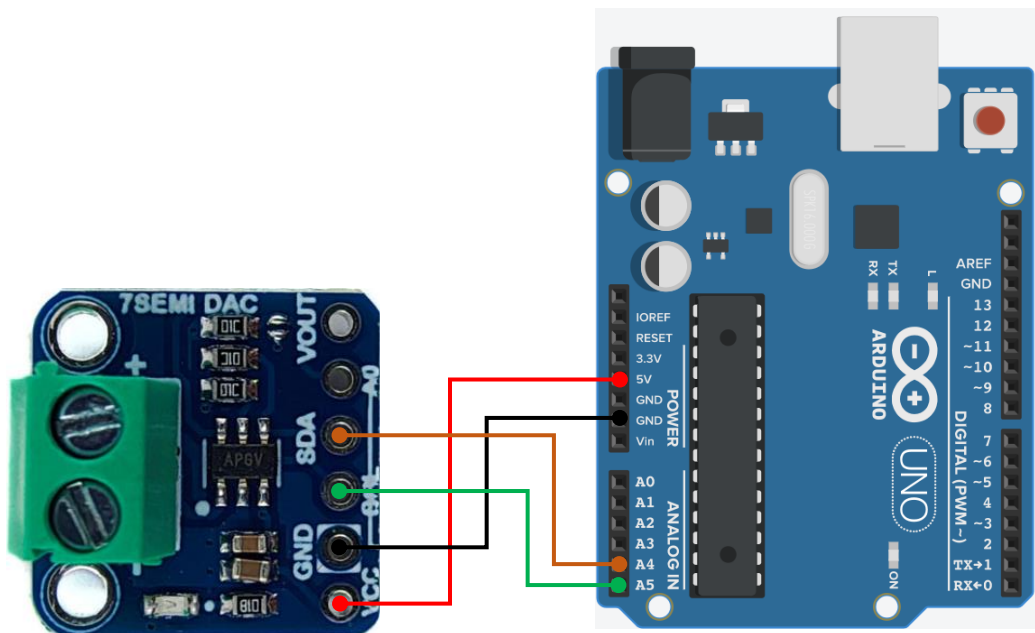
One more basic example could be:-

Let's say you want to set three reference voltages in your project, doing it manually by turning the pot will take time and could also be not precise adjustment. In that case you can give three separate commands to MCP4725 board using some microcontroller to set it automatically.

### Pinouts



### Connections with Arduino UNO



- At first we will check the I2C address of the board, for that copy the code in the given link and paste it in your Arduino IDE.

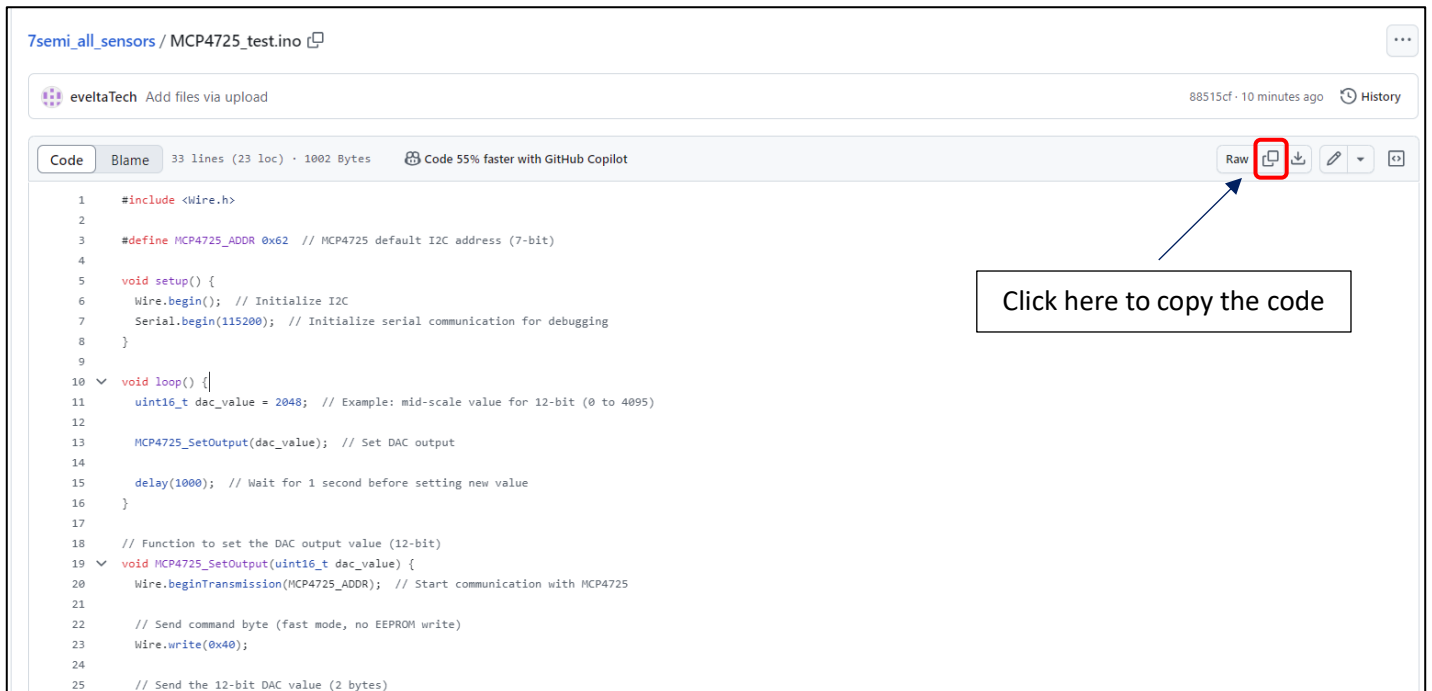
Code link:- [I2C scanner code](#)

You should see the default I2C address for the board .

**“0x62”** or whichever will be the default address.

- To test the board with a sample code, follow the steps below:-

Step-1:- Click on the link given and copy the code- [MCP4725 test code](#)



```
7semi_all_sensors / MCP4725_test.ino
eveltaTech Add files via upload 88515cf · 10 minutes ago History
Code Blame 33 lines (23 loc) · 1002 Bytes Code 55% faster with GitHub Copilot Raw
1 #include <Wire.h>
2
3 #define MCP4725_ADDR 0x62 // MCP4725 default I2C address (7-bit)
4
5 void setup() {
6   Wire.begin(); // Initialize I2C
7   Serial.begin(115200); // Initialize serial communication for debugging
8 }
9
10 void loop() {
11   uint16_t dac_value = 2048; // Example: mid-scale value for 12-bit (0 to 4095)
12
13   MCP4725_SetOutput(dac_value); // Set DAC output
14
15   delay(1000); // Wait for 1 second before setting new value
16 }
17
18 // Function to set the DAC output value (12-bit)
19 void MCP4725_SetOutput(uint16_t dac_value) {
20   Wire.beginTransmission(MCP4725_ADDR); // Start communication with MCP4725
21
22   // Send command byte (fast mode, no EEPROM write)
23   Wire.write(0x40);
24
25   // Send the 12-bit DAC value (2 bytes)
```

Step-2:- To vary the output analog voltage there is a range of ‘0’ to ‘4095’ which you have to set in your code.

If you give VCC = 3.3V, 0 --> 0 volts and 4095 --> 3.3V

If you give VCC = 5V, 0 --> 0 volts and 4095 --> 5V

So, you can set any voltage between 0V – 5V for your application.

Step-3:- In the code **“uint16\_t dac\_value = 2048”** means we want the mid analog value of the input voltage we have provided.

Upload the code and with the help of multimeter check the voltage at **VOUT** with respect to **GND**.

The multimeter should read **1.65V** if your input is **3.3V**.

The multimeter should read **2.5V** if your input is **5V**.