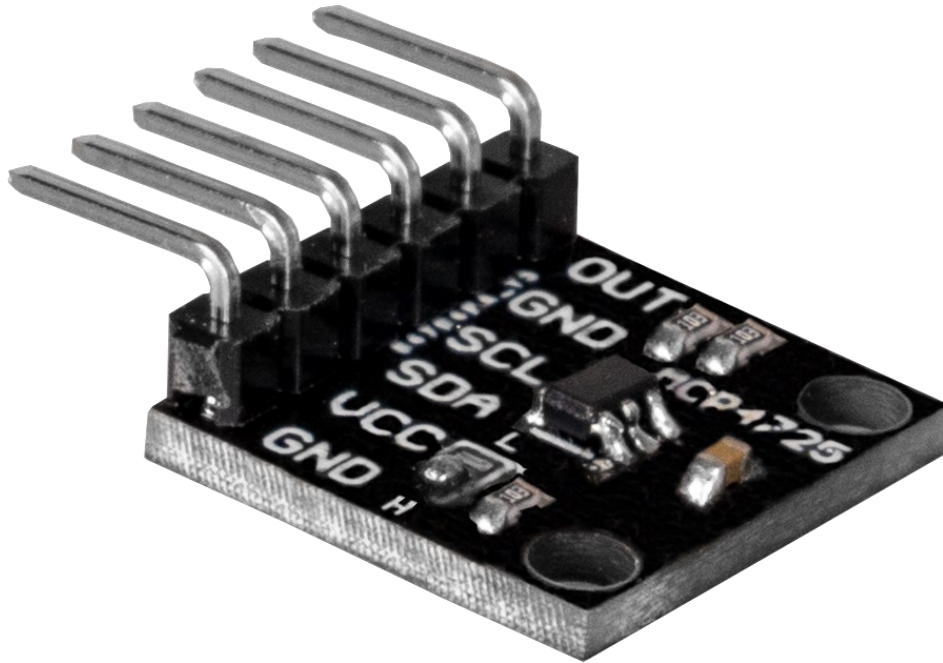# DIGITAL- ANALOG CONVERTER
COM-DAC01

1. GENERAL INFORMATION

Dear customer,
thank you very much for choosing our product.
In the following, we will introduce you to what to observe while starting up and using this product.
Should you encounter any unexpected problems during use, please do not hesitate to contact us.

2. CHANGE I2C ADDRESS

You can change the I2C address of this converter by means of a soldering bridge. Thereby, the address will change from 0x60 to 0x61 after soldering.
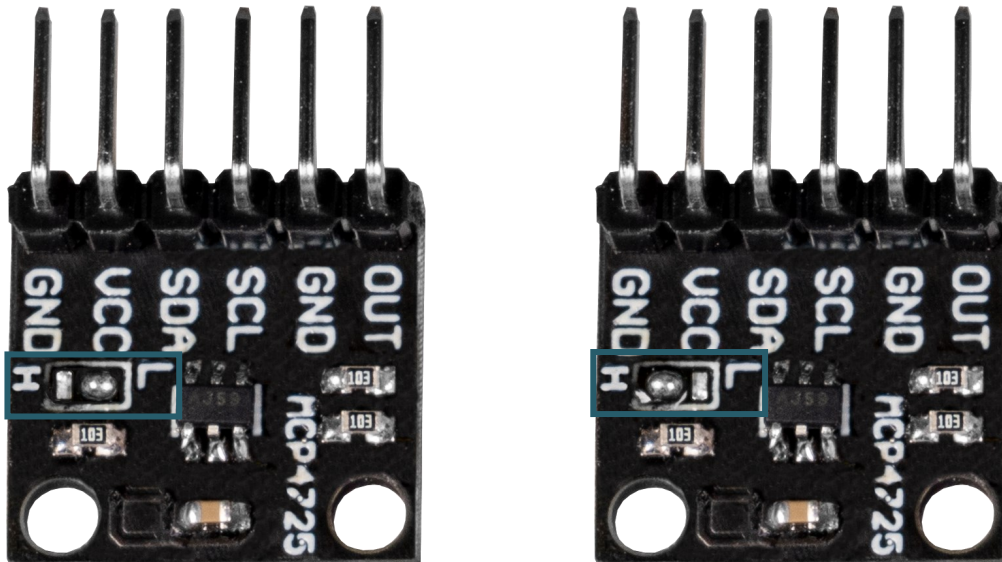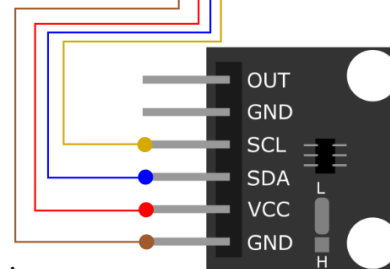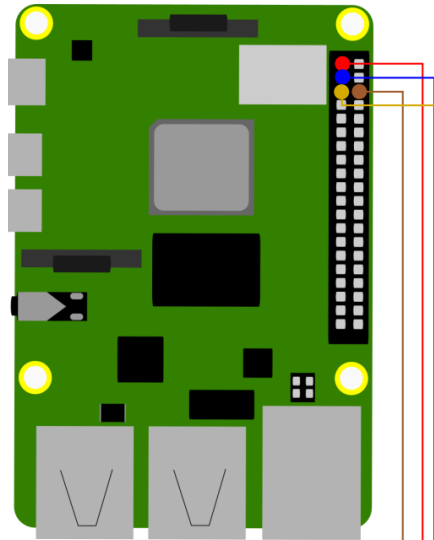
Soldering bridge	Address
L	0x60
H	0x61

> **i** This guide was written under Raspberry Pi OS Bookworm for the Raspberry Pi 4 and 5. It has not been checked with newer operating systems or hardware.

3.1 Connection

COM-DAC01	Raspberry Pi
OUT	-
GND	-
SCL	GPIO 3 (SCL)
SDA	GPIO 2 (SDA)
VCC	3.3 V
GND	GND

The analog output of the converter is via the **OUT** pin.

3.2 Code example

First enable the I2C interface of your Raspberry Pi. To do this, enter the following command in your terminal.

```
sudo raspi-config
```

Select Interface Options → I2C and enable the I2C interface.

After activating the interface, you must set up the virtual environment required for this project. To do this, enter the following commands:

```
mkdir project-name && cd project-name
```

```
python -m venv --system-site-packages env
```

```
source env/bin/activate
```

To use the digital-to-analog converter, we use the library [Adafruit CircuitPython MCP4725](#), which was published by [Adafruit](#) under the [MIT license](#).
Now execute the following command to install this library.

```
sudo apt install python3-pip
```

```
pip3 install adafruit-circuitpython-mcp4725
```

Now restart your Raspberry Pi with the following command:

```
sudo reboot
```

After the restart, you must first switch back to the previously created directory and restart the virtual environment. To do this, enter the following commands:

```
cd project-name
```

```
source env/bin/activate
```

Now you can download the code example with the following command. This is the code example of the Adafruit library but adapted to our COM-DAC01 module.

```
wget https://joy-it.net/files/files/Produkte/COM-DAC01/COM-DAC01.zip
```

Unzip the file and remove the .zip folder with the following command:

```
unzip COM-DAC01.zip && rm COM-DAC01.zip
```

In the example, the I2C address of the module is **0x60**. To change the address, enter the following command:

```
nano COM-DAC01.py
```

For the I2C address **0x60**, the code must look like this:

```
15 # Initialize MCP4725 with address 0x61
16 #dac = adafruit_mcp4725.MCP4725(i2c,address=0x61)
17
18 # Initialize MCP4725 with address 0x60
19 dac = adafruit_mcp4725.MCP4725(i2c,address=0x60)
```

And for the I2C address **0x61**, the code must look like this:

```
15 # Initialize MCP4725 with address 0x61
16 dac = adafruit_mcp4725.MCP4725(i2c,address=0x61)
17
18 # Initialize MCP4725 with address 0x60
19 #dac = adafruit_mcp4725.MCP4725(i2c,address=0x60)
```

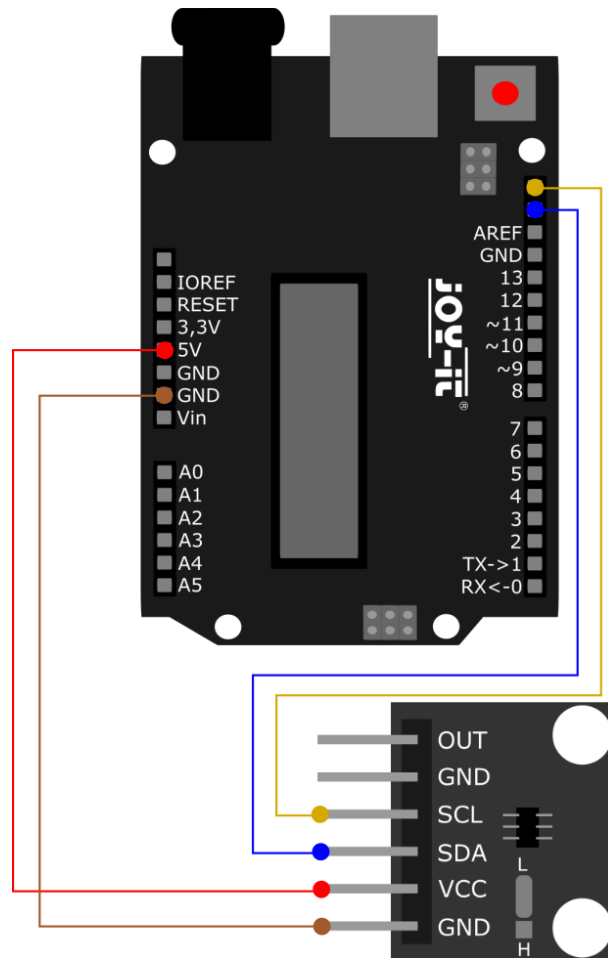Now save the file with **CTRL + O** and **Enter** and close the file with **CTRL + X**.

You can now execute the sample code with the following command:

```
python3 COM-DAC01.py
```

4.1 Connection

COM-DAC01	Arduino
OUT	-
GND	-
SCL	A4
SDA	A5
VCC	5 V
GND	GND

The analog output of the converter is at the pin **OUT**.

4.2 Code example

For this converter, we use the **Adafruit_MCP4725** library from **Adafruit**, which was released under the *BSD license* . You can find and install this in your Arduino IDE under **Sketch → Include Library → Manage Libraries…** . After installation, you can find the **sinewave** example file under **File → Examples → Adafruit MCP4725**. However, before you can run it, you have to adjust the I2C address in line 181.

```
174  void setup(void) {
175    Serial.begin(9600);
176    Serial.println("Hello!");
177
178    // For Adafruit MCP4725A1 the address is 0x62 (default) or 0x63 (ADDR pin tied to VCC)
179    // For MCP4725A0 the address is 0x60 or 0x61
180    // For MCP4725A2 the address is 0x64 or 0x65
181    dac.begin(0x60);
182
183    Serial.println("Generating a sine wave");
184  }
```

Now, you can run your program by clicking the **Upload** button. However, before doing so, make sure that the **Port** and **Board** are correctly selected under **Tools**.

5. ADDITIONAL INFORMATION

Our information and take-back obligations according to the Electrical and Electronic Equipment Act (ElektroG)

Symbol on electrical and electronic equipment:

This crossed-out dustbin means that electrical and electronic appliances do not belong in the household waste. You must return the old appliances to a collection point.
Before handing over waste batteries and accumulators that are not enclosed by waste equipment must be separated from it.

Return options:
As an end user, you can return your old device (which essentially fulfils the same function as the new device purchased from us) free of charge for disposal when you purchase a new device.
Small appliances with no external dimensions greater than 25 cm can be disposed of in normal household quantities independently of the purchase of a new appliance.

Possibility of return at our company location during opening hours:
SIMAC Electronics GmbH, Pascalstr. 8, D-47506 Neukirchen-Vluyn, Germany

Possibility of return in your area:
We will send you a parcel stamp with which you can return the device to us free of charge. Please contact us by email at Service@joy-it.net or by telephone.

Information on packaging:
If you do not have suitable packaging material or do not wish to use your own, please contact us and we will send you suitable packaging.

6. SUPPORT

If there are still any issues pending or problems arising after your purchase, we will support you by e-mail, telephone and with our ticket support system.

Email: **service@joy-it.net**
Ticket system: https://support.joy-it.net
Telephone: +49 (0)2845 9360-50 (Mon - Thur: 09:00 - 17:00 o'clock CET,

Fri: 09:00 - 14:30 o'clock CET)

For further information please visit our website:
www.joy-it.net

Published: 2024.04.18