Arduino - MCP49xx (DAC)

Hans-Petter Halvorsen

DAC

- DAC – Digital to Analog Converter

- Why? Arduino UNO has no Analog Out Channels.
- We can then use the SPI bus (or I2C bus) available on the Arduino together with a commercial DAC chip in order to create our own Analog Out signal

MCP49xx

- A product line with different DAC chips with different technical specifications

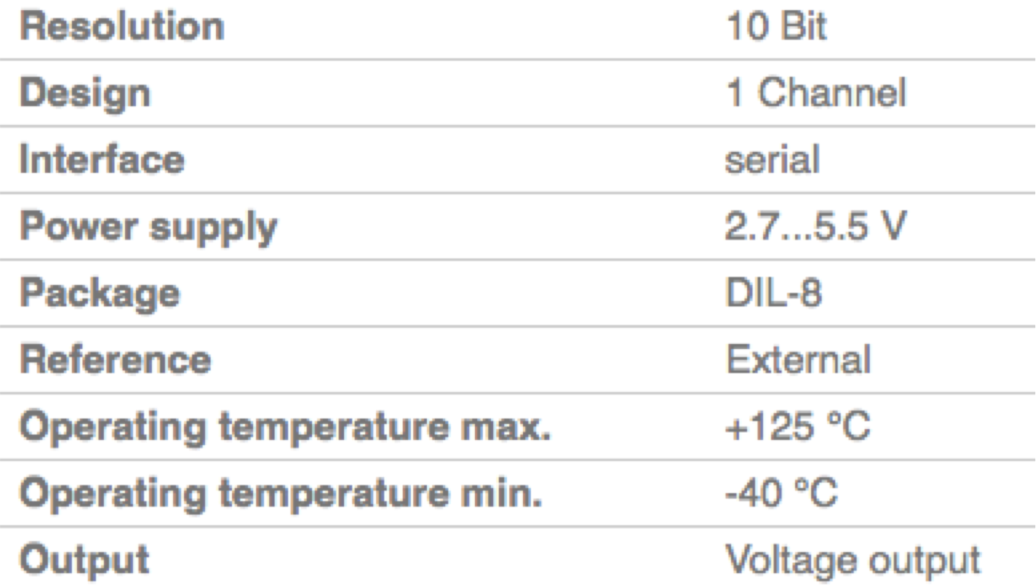
MCP4901: 8-Bit Voltage Output DAC
MCP4911: 10-BitVoltage Output DAC
MCP4921: 12-Bit Voltage Output DAC

MCP4911

MCP4911

- The MCP4911 Microchip is a single channel 10-bit digital analog channel (DAC) with an external voltage reference.
- The communication between the MCP4911 and the Arduino microcontroller is done through the Serial Peripheral Interface (SPI) protocol

MCP4911 Technical Specifications



Resolution	10 Bit
Design	1 Channel
Interface	serial
Power supply	2.7...5.5 V
Package	DIL-8
Reference	External
Operating temperature max.	+125 °C
Operating temperature min.	-40 °C
Output	Voltage output

Datasheet: https://www.elfadistrelec.no/Web/Downloads/he/et/MCP4901_11_eng_datasheet.pdf

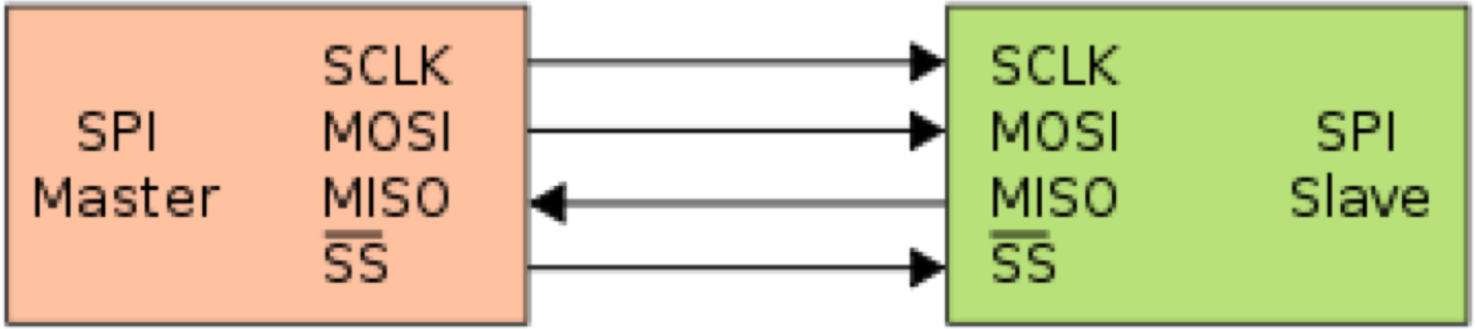
MCP4911

Where to buy?

Can be bought many places, e.g., ELFA DISTRELEC

https://www.elfadistrelec.no/en/converter-ic-10-bit-dil-microchip-mcp4911/p/17381762

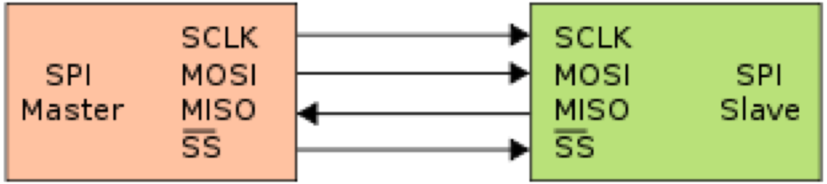
SPI Bus


SPI
Master
SCLK
MOSI
MISO
$\overline{SS}$
SCLK
MOSI
MISO
$\overline{SS}$
SPI
Slave

SPI Bus

- Serial Peripheral Interface (SPI) is a synchronous serial data protocol used by microcontrollers for communicating with one or more peripheral devices quickly over short distances.
- With an SPI connection there is always one master device (usually a microcontroller) which controls the peripheral devices.
- SPI devices communicate in full duplex mode using a master-slave architecture with a single master.
- The interface was developed by Motorola and has become a de facto standard.
- Typical applications include sensors, Secure Digital cards, and liquid crystal displays (LCD).

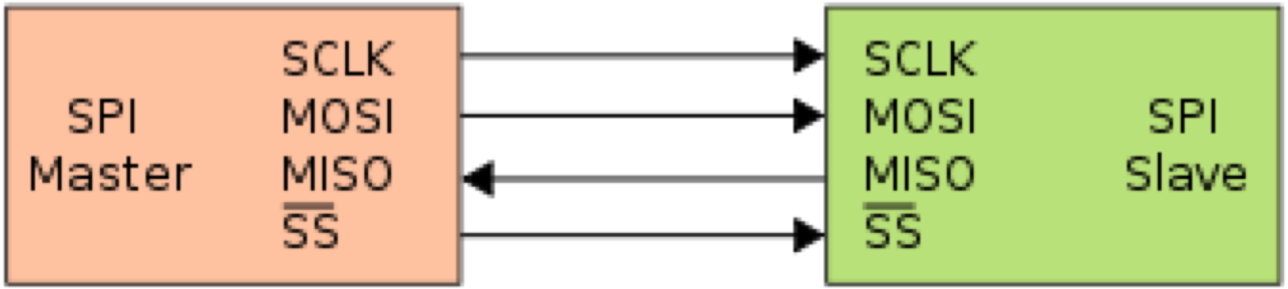
SCLK : Serial Clock (output from master)
MOSI : Master Output, Slave Input (output from master)
MISO : Master Input, Slave Output (output from slave)
SS (or SC) : Slave Select (active low, output from master)



http://en.wikipedia.org/wiki/Serial_Peripheral_Interface_Bus

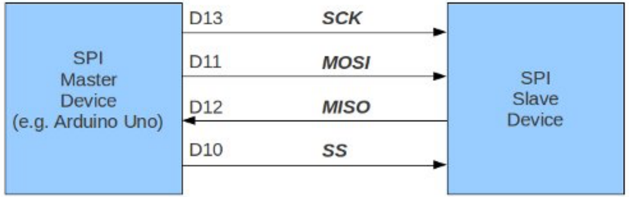
https://learn.sparkfun.com/tutorials/serial-peripheral-interface-spi

SPI Bus Example



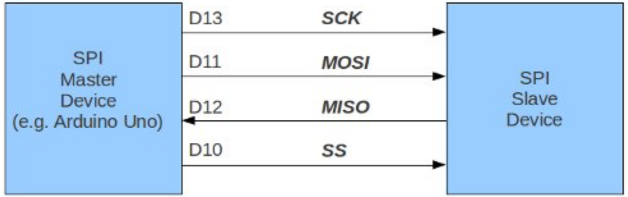
Arduino
is a SPI Master

The MCP4911 DAC
chip is a SPI slave



Arduino SPI

- https://www.arduino.cc/en/Reference/SPI
- http://tronixstuff.com/2011/05/13/tutorial-arduino-and-the-spi-bus/
- http://arduino.stackexchange.com/questions/16348/how-do-you-use-spi-on-an-arduino
- https://learn.sparkfun.com/tutorials/serial-peripheral-interface-spi



SPI References

- SPI, Wikipedia: https://en.wikipedia.org/wiki/Serial_Peripheral_Interface_Bus
- SPI, Sparkfun: https://learn.sparkfun.com/tutorials/serial-peripheral-interface-spi

Arduino SPI Library

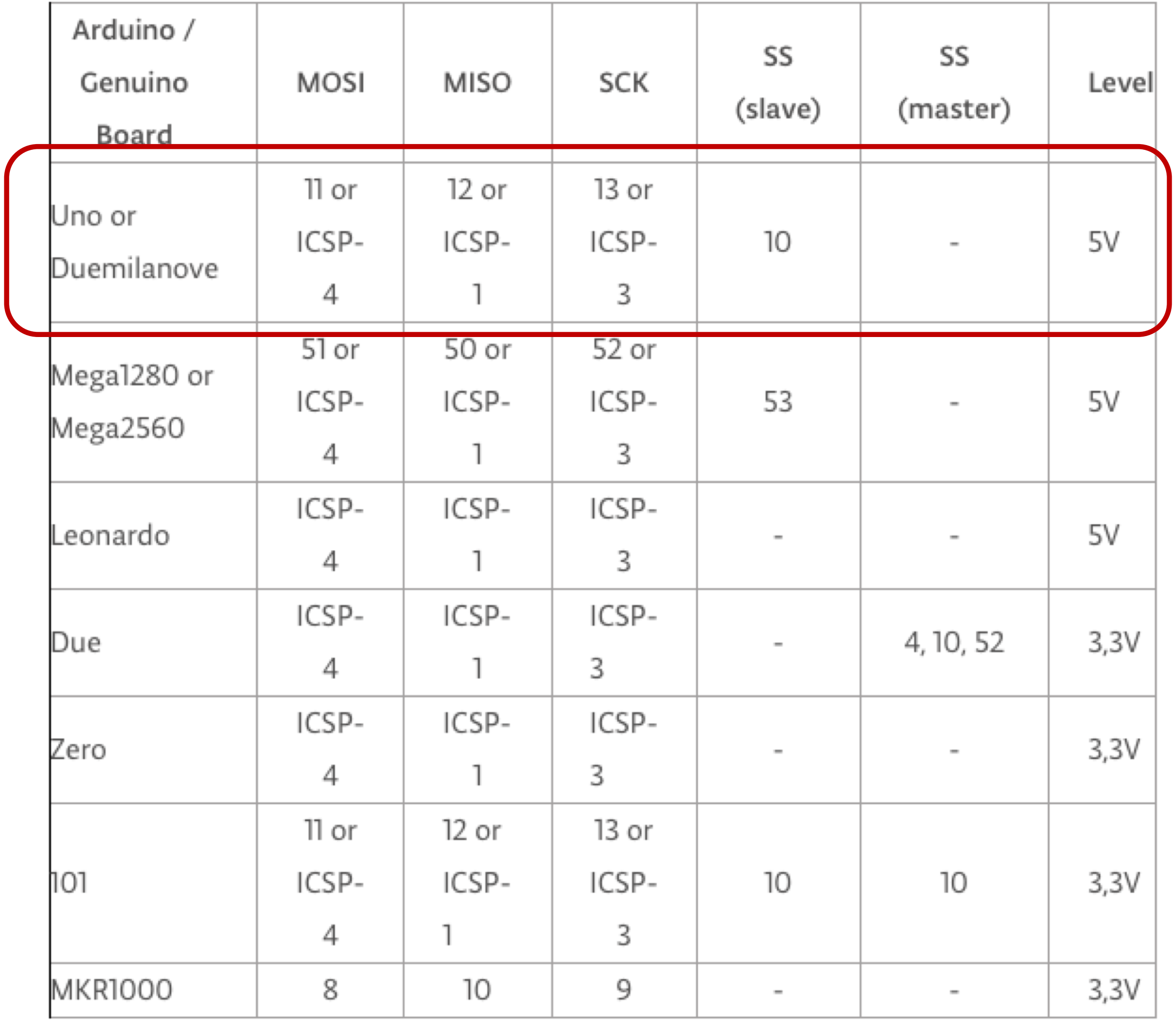
- https://www.arduino.cc/en/Reference/SPI

Arduino + MCP49xx

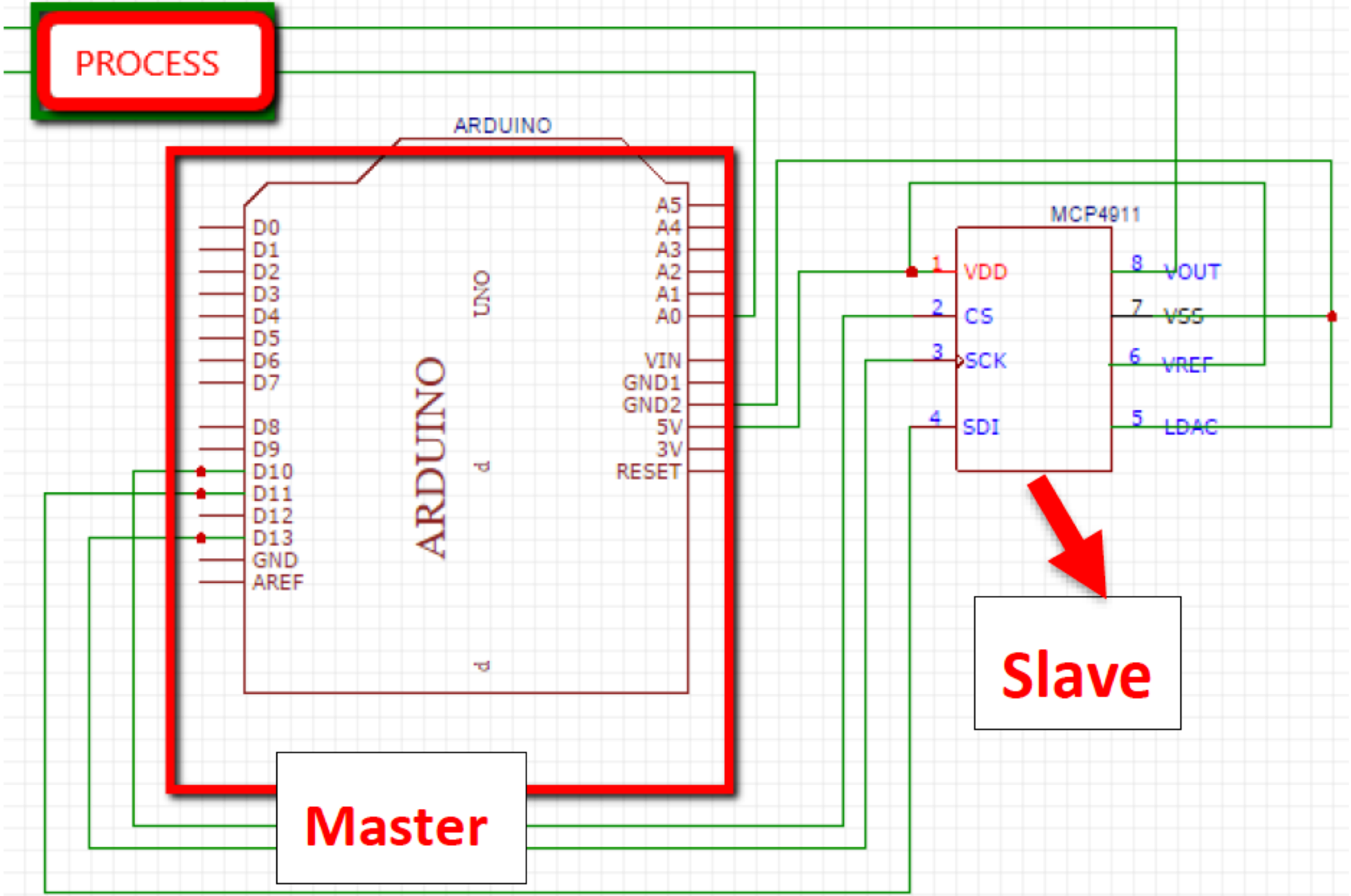
How?

1. We can use the Arduino SPI Library. A library that can be used together with kinds of devices supporting the SPI bus. https://www.arduino.cc/en/Reference/SPI
2. Or even easier, we can use the MCP49xx Arduino Library. Here we have a premade functions that are tailor-made for the MCP49xx chips. https://github.com/exscape/electronics/tree/master/Arduino/Libraries

Arduino SPI Connections



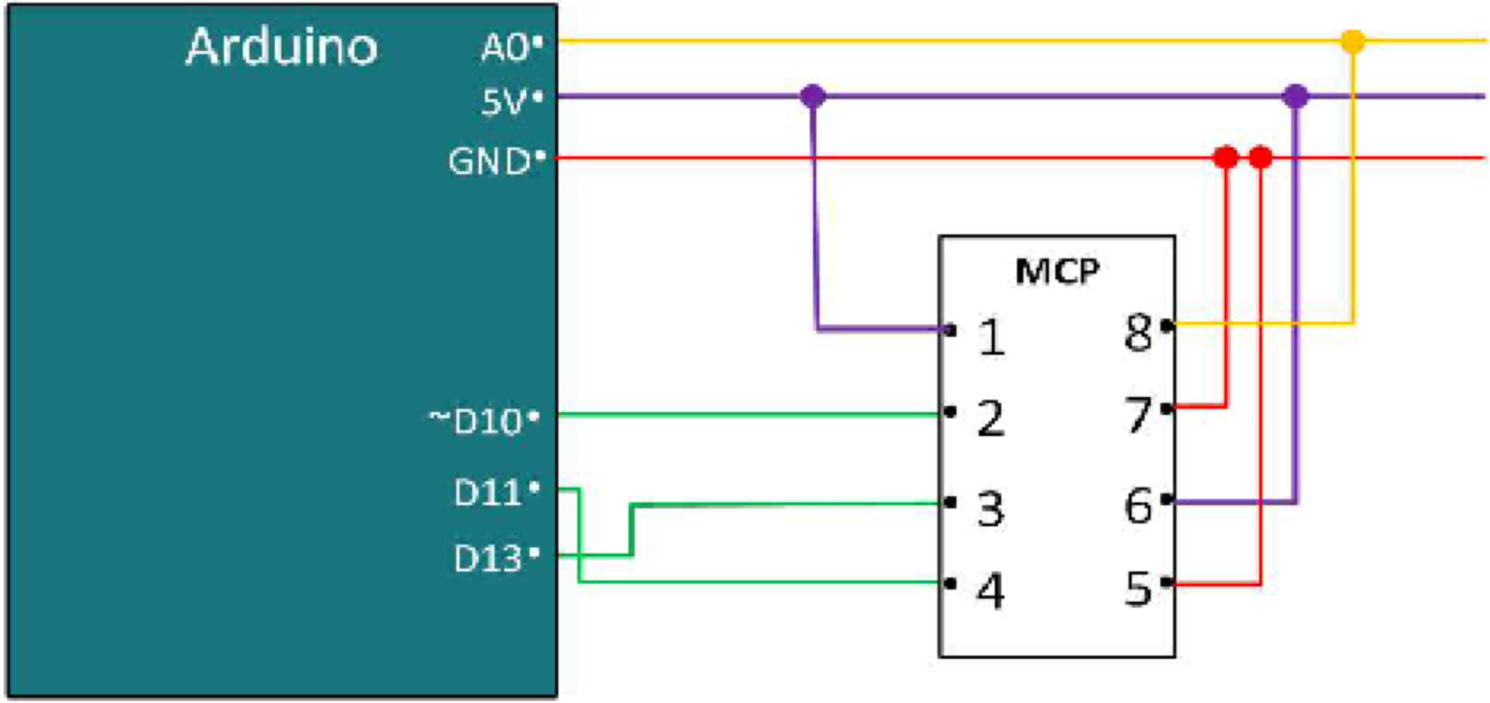
Arduino / Genuino Board	MOSI	MISO	SCK	SS (slave)	SS (master)	Level
Uno or Duemilanove	11 or ICSP-4	12 or ICSP-1	13 or ICSP-3	10	-	5V
Mega1280 or Mega2560	51 or ICSP-4	50 or ICSP-1	52 or ICSP-3	53	-	5V
Leonardo	ICSP-4	ICSP-1	ICSP-3	-	-	5V
Due	ICSP-4	ICSP-1	ICSP-3	-	4, 10, 52	3,3V
Zero	ICSP-4	ICSP-1	ICSP-3	-	-	3,3V
101	11 or ICSP-4	12 or ICSP-1	13 or ICSP-3	10	10	3,3V
MKR1000	8	10	9	-	-	3,3V



PROCESS

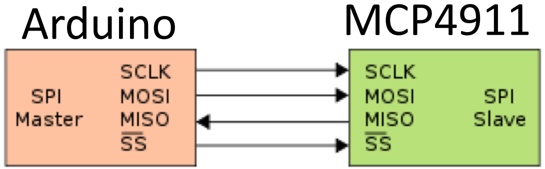
Master

Slave


Arduino
A0
5V
GND
~D10
D11
D13
MCP
1
2
3
4
8
7
6
5

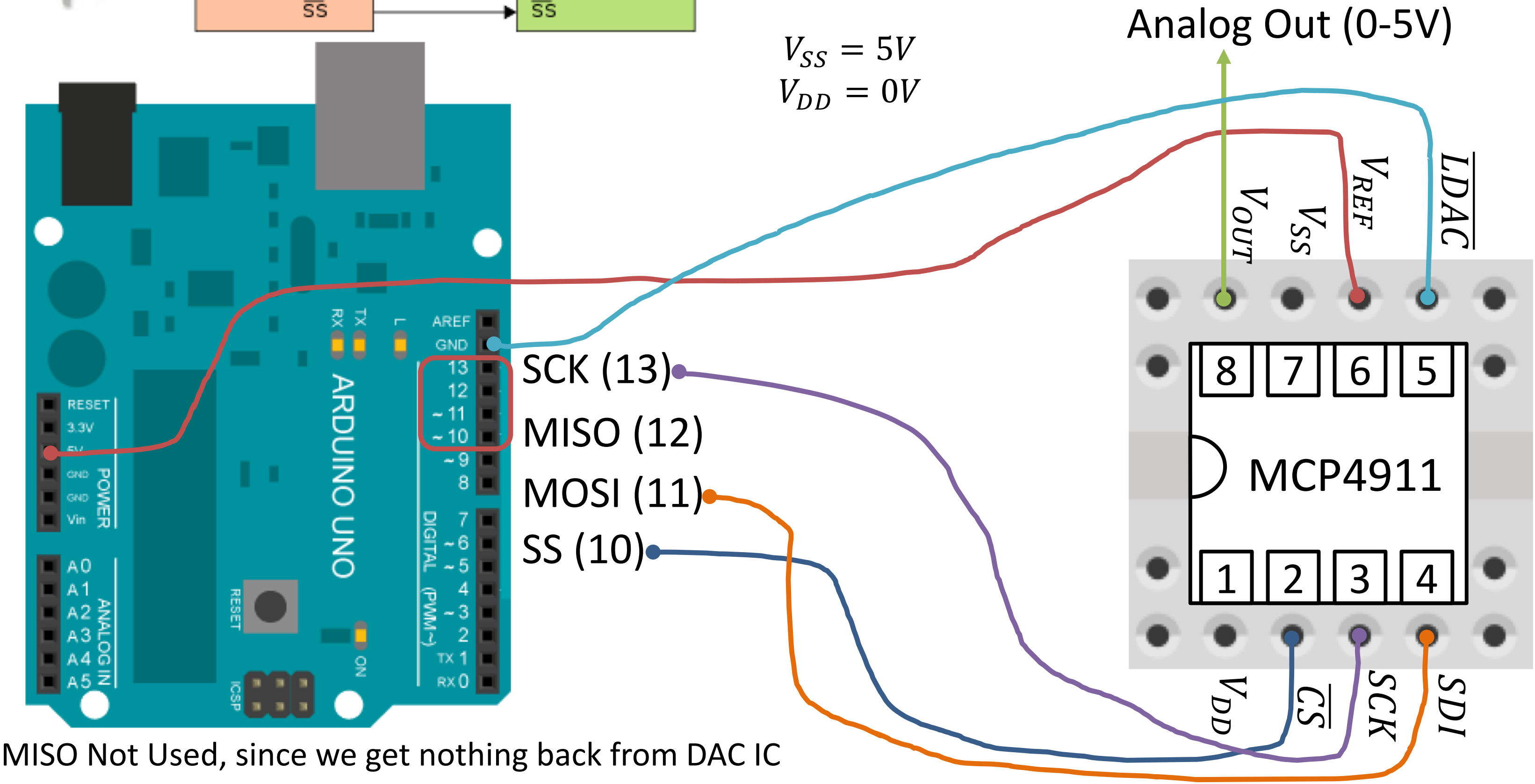
DAC

MCP4911: 10-bit single DAC



The LDAC input can be used to select the device, and you could use a GPIO pin to turn the device on and off through this pin. In this example, we just tie it to ground so it is always selected and powered.

$$V_{SS} = 5V$$
$$V_{DD} = 0V$$



MISO Not Used, since we get nothing back from DAC IC

MCP49xx Arduino Library Example

```
#include <SPI.h>
#include <DAC_MCP49xx.h>

// The Arduino pin used for the slave select / chip select
#define SS_PIN 10

//Set up the DAC
DAC_MCP49xx dac(DAC_MCP49xx::MCP4911, SS_PIN);

void setup()
{

}

void loop()
{
  // For MCP4911, use values below (but including) 1023 (10 bit)
  dac.output(255);
  delay(2500);

  dac.output(0);
  delay(2500);
}
```

https://github.com/exscape/electronics/tree/master/Arduino/Libraries

MCP49xx Arduino Library Example

```
#include <SPI.h>           //Include the Arduino SPI Library
#include <DAC_MCP49xx.h>   //Include the MCP49xx Arduino Library

// The Arduino pin used for the slave select / chip select
#define SS_PIN 10

DAC_MCP49xx dac(DAC_MCP49xx::MCP4911, SS_PIN);

void setup()
{
  Serial.begin(9600);
}

void loop()
{
  double u; //Control Signal
  int aiPin = 0;
  int aiValue;

  for (int i=0; i<1023; i++)
  {
    u = i;
    dac.output(u);

    aiValue = analogRead(aiPin);
    Serial.print("AIValue=");
    Serial.println(aiValue);

    delay(1000);
  }
}
```

Connect the circuit (Arduino + MCP4911) on a breadboard. Use a multi-meter so see if you get the correct output signal.

On the Multimeter you should see the output slowly increasing from ~0V to ~5V with intervals of 1000ms.

You can also connect the output from the DAC to an Analog Input Pin on the Arduino. Write the value to the Serial Monitor.

References

- Arduino SPI Library: https://www.arduino.cc/en/Reference/SPI
- MCP49xx Arduino Library: https://github.com/exscape/electronics/tree/master/Arduino/Libraries
- SPI: http://en.wikipedia.org/wiki/Serial_Peripheral_Interface_Bus
- SPI: https://learn.sparkfun.com/tutorials/serial-peripheral-interface-spi

Hans-Petter Halvorsen

University of South-Eastern Norway

www.usn.no

E-mail: hans.p.halvorsen@usn.no

Web: http://www.halvorsen.blog