Introduction to Soft Robotics

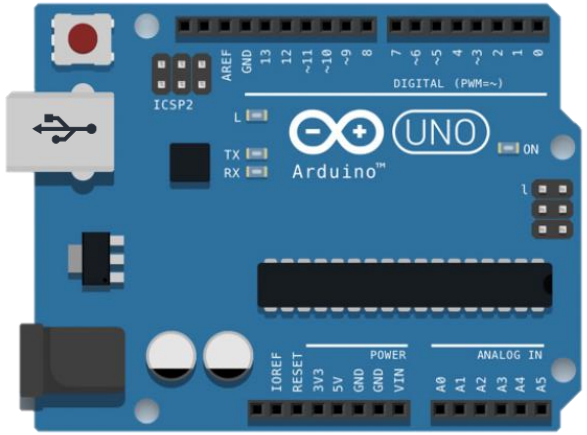
Autumn 2022

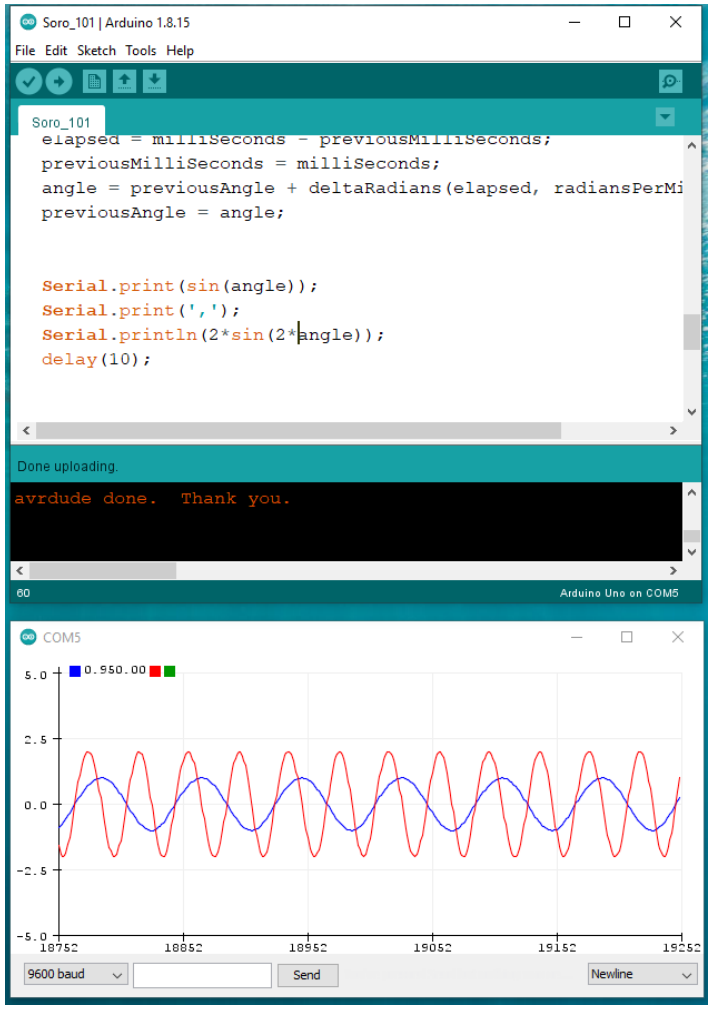
Instructors: Ahmad Rafsanjani, Jonas Jørgensen

Tutorial 1: Arduino 101

5 September 2022

Arduino Uno and Serial communication

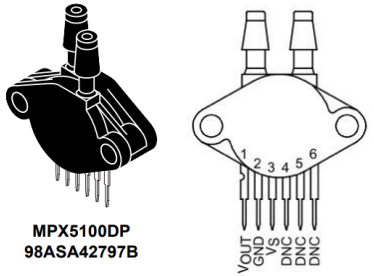




SoRo_101 QR code

Tutorial 2: MPX5100 Integrated Silicon Pressure Sensor

MPX5100, 0 to 100 kPa, Differential, Gauge, and Absolute, Integrated, Pressure Sensor



V_{out}	Output Voltage
GND	Ground
V_s	Voltage Supply
DNC	Do not connect

Characteristic	Symbol	Min	Typ	Max	Unit
Pressure range[1] Gauge, differential: MPX5100G/MPXV5100G Absolute: MPX5100AP	P_{OP}	 0 15	 — —	 100 115	kPa
Supply voltage[2]	V_S	4.75	5.0	5.25	V_{DC}
Supply current	I_O	—	7.0	10	mAdc

Figure 1. MPX5100DP Pinout (top view), Pin functions, and mechanical and electrical specifications

Nominal Transfer Value: $V_{OUT} = V_S (P \times 0.009 + 0.04)$
$\pm$ (Pressure Error x Temp. Mult. x 0.009 x V_S)
$V_S = 5.0\ V \pm 0.25\ V$

Figure 8. Transfer function (MPX5100D, MPX5100G, MPXV5100G)

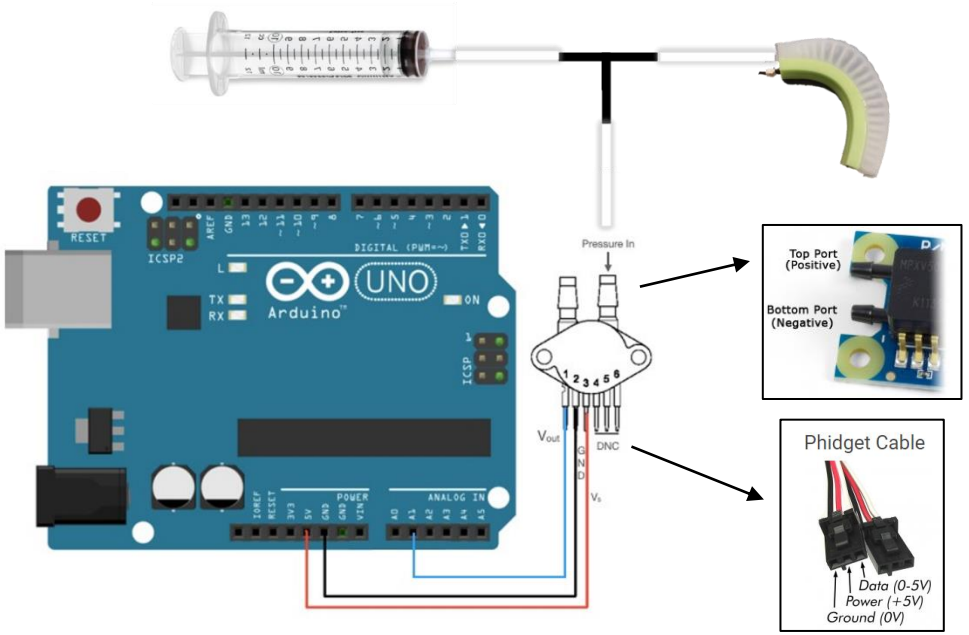


Figure 2. MPX5100DP – Arduino connection diagram

Code Information

Arduino libraries

→ Download ads Arduino driver at GitHub Link
→ Copy folder ***ads_driver*** into ***Arduino/Libraries*** folder
→ Open the ***ads.h*** file located at ***ads_driver*** folder
→ Change the command ADS_DFU_CHECK(1) to ADS_DFU_CHECK(0)
→ Save the changes
→ Download ***SoRo_Tutorial_1*** sketch at GitHub Link

Ads_driver QR code

Arduino Calibration

→ Compile and Upload ***SoRo_Tutorial_1*** sketch at Arduino Uno Device
→ Open the serial monitor
→ Put the Bendlabs sensor in the 0° position
→ Type 0 and press Enter
→ Put the Bendlabs sensor in the 90° position
→ Type 9 and press Enter

SoRo_Tutorial_1 QR code

Code Information

Arduino Calibration and plotting

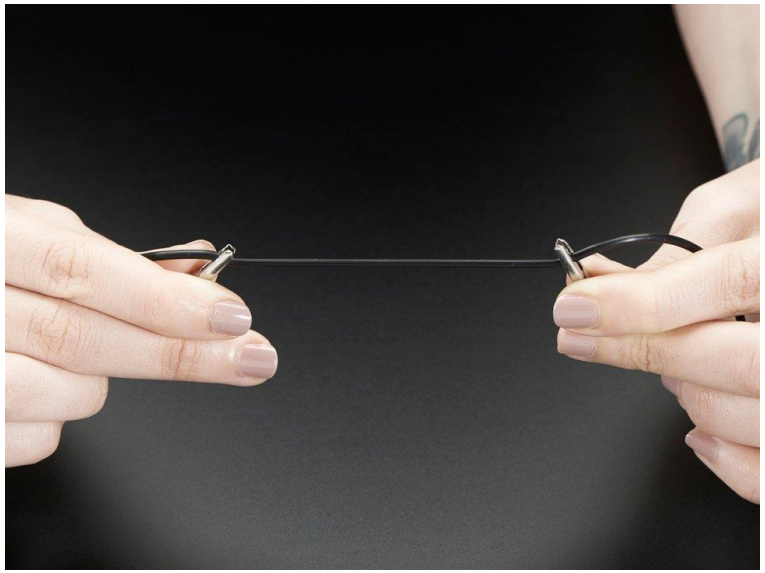
```
//Pressure sensor calibration factors  MPX5100 Series Integrated Silicon Pressure Sensor analog input (0 to 100 kPa)  Vout=Vs(P * 0.009 + 0.04),  Vs=5V = 1024,  P =

const float SensorOffset = 4.44;  //pressure sensor offset
const float SensorGain = 0.109;   // pressure sensor proportional relation
```

```
// read the input on analog pin 1:
float pressure_sensorValue = (analogRead(PRESSURE_SENSOR)*SensorGain-SensorOffset); //Do maths for calibration
```

```
Serial.print(sample[0]);    // Angle data
Serial.print(",");
Serial.print(pressure_sensorValue);    // pressure data in kpa
Serial.print(",");
Serial.println(resistance_sensorValue);  // Stretch data
```

Tutorial 3: Conductive Rubber Cord Sensor



TECHNICAL DETAILS

Length: approximately 1 meter = 39 inches
Diameter: 2mm
Resistance: 350-400 ohms per inch / 140 - 160 ohms per centimeter

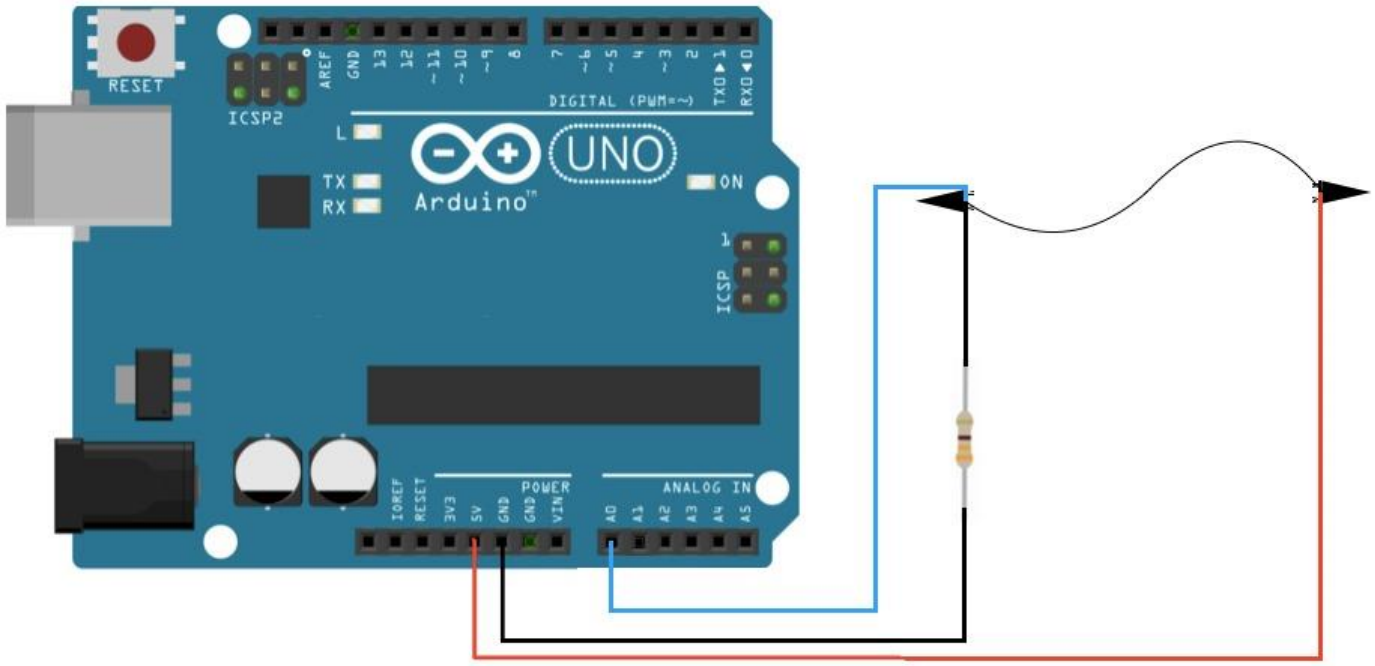


Figure 1. Conductive Rubber Cord Sensor – Arduino connection diagram

Code Information

Arduino Calibration and plotting

```
//Pressure sensor calibration factors

const float SensorOffset2 = 330;  //pressure sensor offset
const float SensorGain2 = 337590;   // pressure sensor proportional relation
```

```
// read the input on analog pin 0:
float resistance_sensorValue = (SensorGain2/analogRead(RUBBER_SENSOR)-SensorOffset2); //Do maths for calibration
```

```
Serial.print(sample[0]);    // Angle data
Serial.print(",");
Serial.print(pressure_sensorValue);    // pressure data in kpa
Serial.print(",");
Serial.println(resistance_sensorValue);  // Stretch data
```

Tutorial 4: Bendlabs sensor

bendlabs

One Axis

Sensor Specifications

- Dimensions: 100mm x 7.62mm x 1.27mm (3.94in x 0.30in x 0.05in)
- Average Sensitivity: 0.274 pF/°
- Repeatability: 0.18°
- Life Cycle: >1M cycles

Electrical Specifications

- Sensitivity: 0.016° LSB
- Voltage: 1.62 - 3.63V
- Output: I²C

4-channel Logic Level Converter

Figure 1. Bendlabs sensor – Logic Level Converter – Arduino connection diagram

Code Information

Arduino Calibration and plotting

```
//BendSensor functions for dataprocessing

void ads_data_callback(float * sample);                    // Bendlabs sensor information callback
void deadzone_filter(float * sample);                      // Bendlabs sensor Deadzone filter
void signal_filter(float * sample);                        // Bendlabs signal filter
void parse_com_port(void);                                 // I2C communication decoding
```

```
//Defining varialbes for bendlabs sensor data processing
static float sample[2];
uint8_t data_type;

// Read data from the one axis ads sensor
int ret_val = ads_read_polled(sample, &data_type);

// Check if read was successfull

  if(data_type == ADS_SAMPLE)
  {
    // Low pass IIR filter
    signal_filter(sample);

    // Deadzone filter
    deadzone_filter(sample);
  }
```

```
  case '0':
    // Take first calibration point at zero degrees
    ads_calibrate(ADS_CALIBRATE_FIRST, 0);
    break;
  case '9':
    // Take second calibration point at ninety degrees
    ads_calibrate(ADS_CALIBRATE_SECOND, 90);
    break;
```

Tutorial 5:
System integration

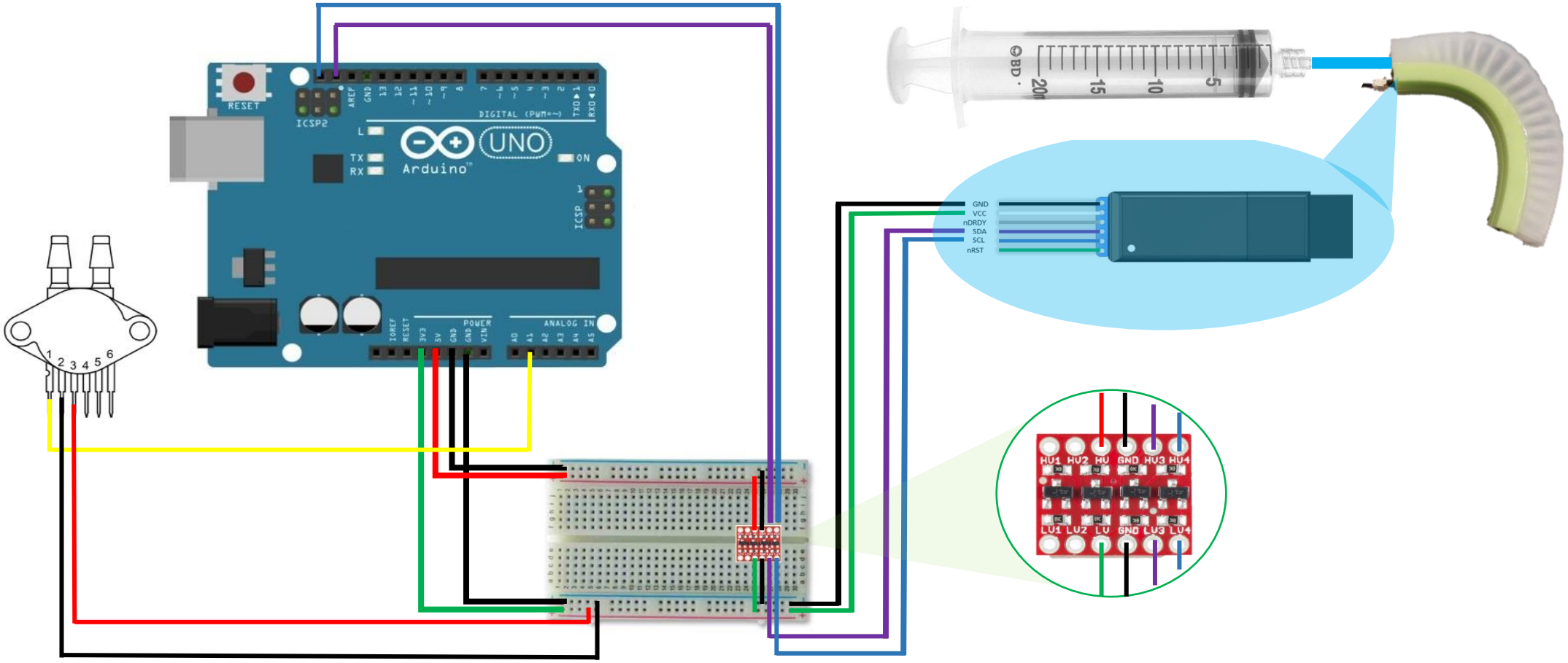


Figure 1. Connection Scheme for a Bendlabs and MPX5100 pressure sensor for Arduino Uno