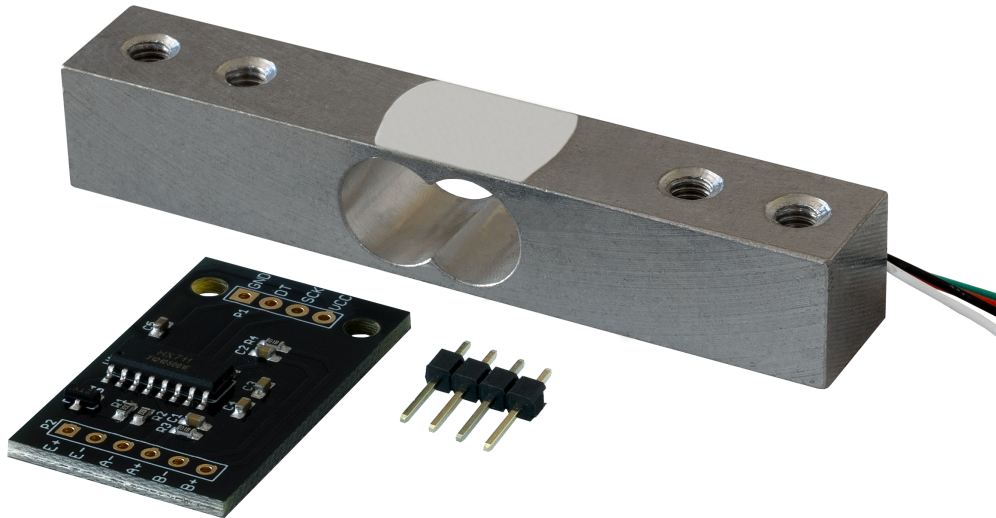


LOAD CELL

SEN-HX711



1. GENERAL INFORMATION

Dear customer,

thank you very much for choosing our product.

In following, we will introduce you to what to observe while starting up and using this product.

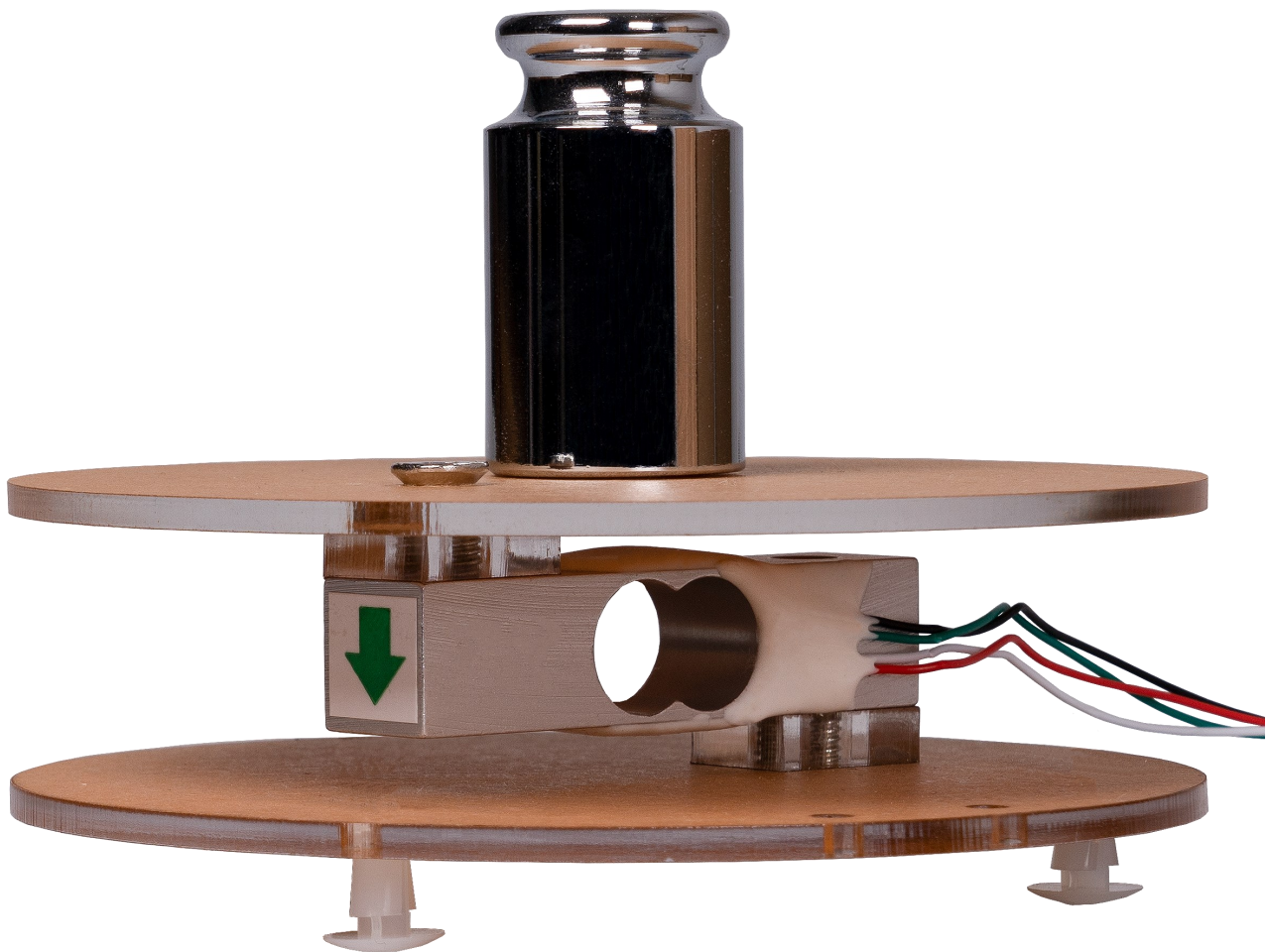
Should you encounter any unexpected problems during use, please do not hesitate to contact us.

2. INSTALLATION OF THE LOAD CELL

For accurate measurements it is important to mount the load cell correctly.

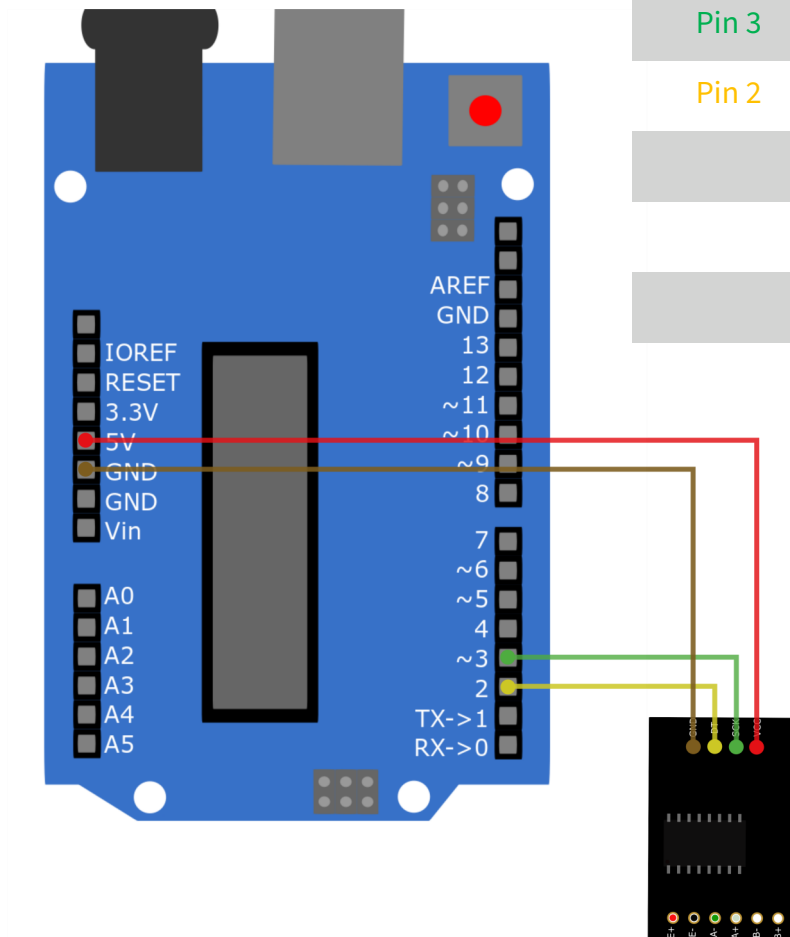
The cell must be bolted into a solid base plate with the Integrated holes on one side and the other side bolted to the bearing surface of your measuring point.

Example setup:



3. USE WITH THE ARDUINO

3.1 Wiring



Arduino	Amplifier board	Load cell
5 V	VCC	-
GND	GND	-
Pin 3	SCK	-
Pin 2	DT	-
	E+	Red
	E-	Black
	A-	Green
	A+	White

3.2 Installation

To use the load cell, you must install a library. To do this, open your Arduino IDE and click on:

Tools -> Manage libraries...

Now enter HX711 Arduino Library in the search field and install the library **HX711 Arduino Library** by Bogdan Necula, Andreas Motl.

3.3 Code example

In the following code example, your load cell will first be calibrated and then repeatedly outputs the currently measured value.

For calibration you need a known weight. Ideally, it should be half of the maximum measurement weight.

The constant Calibration_Weight must be adjusted with the weight you use for calibration. The weight is specified in grams.

```
#include "HX711.h"

// HX711 circuit wiring
const int LOADCELL_DOUT_PIN = 2;
const int LOADCELL_SCK_PIN = 3;
const int Calibration_Weight = 1000;
HX711 scale;

void setup() {
  Serial.begin(57600);
  scale.begin(LOADCELL_DOUT_PIN, LOADCELL_SCK_PIN);
  scale.set_scale();
  scale.tare();
  Serial.println("Calibration");
  Serial.println("Put a known weight on the scale");
  delay(5000);
  float x = scale.get_units(10);
  x = x / Calibration_Weight;
  scale.set_scale(x);
  Serial.println("Calibration finished...");
  delay(3000);
}

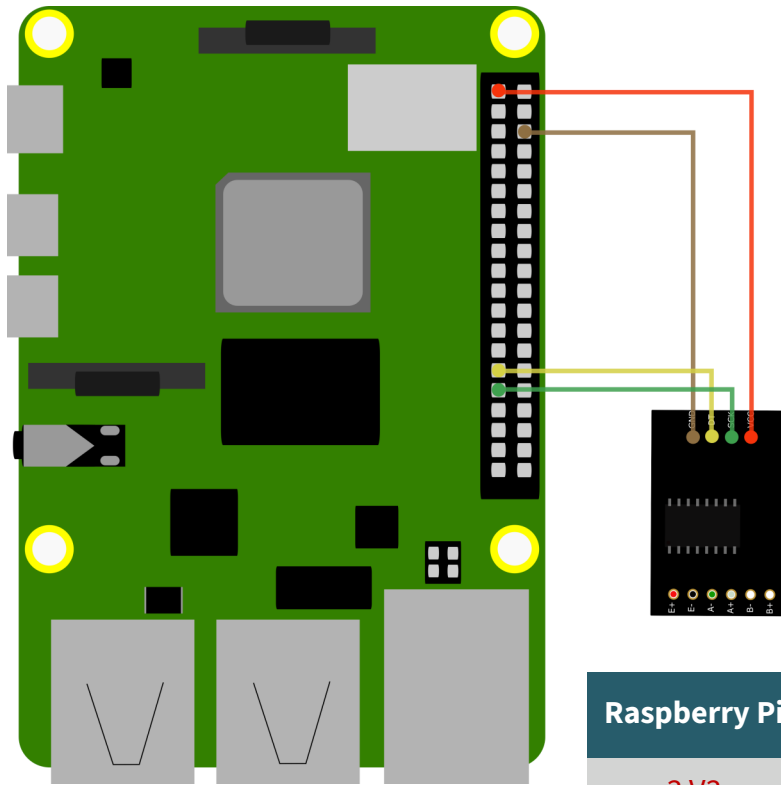
void loop() {

  if (scale.is_ready()) {
    float reading = scale.get_units(10);
    Serial.print("HX711 reading: ");
    Serial.println(reading);
  } else {
    Serial.println("HX711 not found.");
  }

  delay(1000);
}
```

4. USE WITH THE RASPBERRY PI

4.1 Wiring



Raspberry Pi	Amplifier board	Load cell
3 V3	VCC	-
GND	GND	-
GPIO 6	SCK	-
GPIO 5	DT	-
	E+	Red
	E-	Black
	A-	Green
	A+	White

4.2 Installation

To use the load cell, you need a library.
Enter the following commands to create a virtual environment:

```
mkdir sen_hx711
```

```
cd sen_hx711
```

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

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

Now you can install the library with the following command:

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

```
pip3 install JoyIT-hx711py
```

Now you can download the sample codes with the following command:

```
sudo apt install git
```

```
git clone https://github.com/joy-it/JoyIT_hx711py.git
```

4.3 Code example

Now you can start the calibration with the following commands:

```
cd JoyIT_hx711py/examples
```

```
python3 calibration.py
```

For calibration you need a known weight. Ideally, it should be half the maximum measured weight.

Follow the instructions for the calibration that are displayed on the terminal. At the end of the calibration, you can display the values required in the next step by pressing **2** and confirming with **Enter**.

The output should look something like this:

```
Please choose:
[1] Recalibrate.
[2] Display offset and scale and weigh an item!
[0] Clean and exit.
>2

Offset: 8328076.625
Scale: 403.84
Item weighs 100.26077778328052 grams.
```

Make a note of the values Offset and Scale, you must now insert these into your example programme.

To do this, enter the following command:

```
nano example.py
```

Now find the following lines:

```
def setup():
    """
    code run once
    """
    hx.set_offset(8328076.625)
    hx.set_scale(403.84)
```

Now enter your previously noted values for Offset and Scale in the corresponding brackets.
Save the file with the key combination **CTRL + O** and **Enter** and exit the editor with **CTRL + X**.

Now you can execute the sample programme with the following command:

```
python3 example.py
```

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

For further information please visit our website:

www.joy-it.net