

Hmotnostní senzor 0–100 g



POPIS

Senzor může být v kombinaci s vhodným AD převodníkem (např. HX711) použit pro sestavení kuchyňské váhy nebo siloměru.

Základní charakteristika snímače:

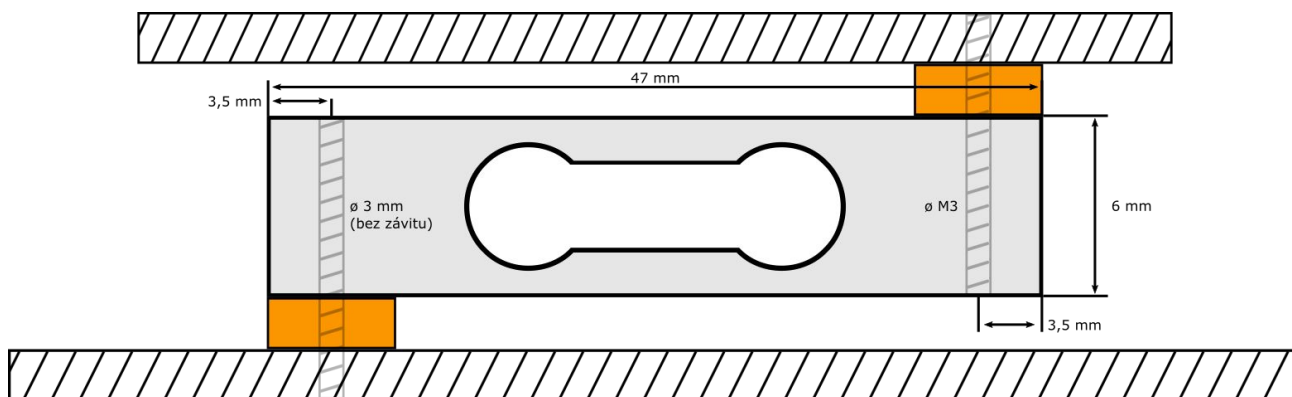
- Max. povolené zatížení 1 N
- Konfigurace: čtveřice tenzometrů zapojených do Wheatstoneova můstku
- Impedance 1 000 Ω
- Snadná montáž díky otvorům se závity



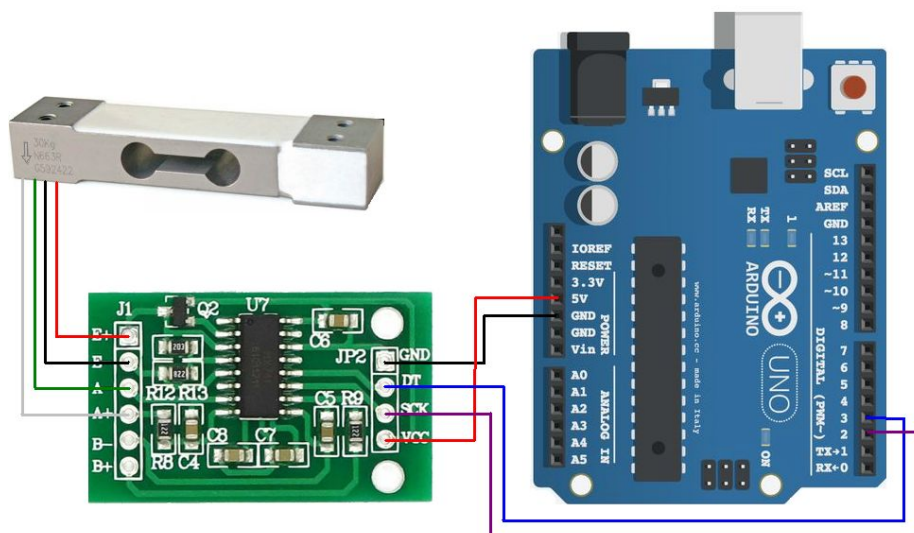
SPECIFIKACE

Měřicí rozsah (N)	0–1	Opakovatelnost (% FS)	0,03
Povolené přetížení (%)	150	Hystereze (% FS)	0,03
Excitační napětí (V DC)	5–15	Tečení (% FS, 30 min)	0,03
Výstupní impedance (Ω)	1 000 \pm 10 %	Stupeň krytí	IP65
Izolační odpor (M Ω)	\geq 300	Rozsah provoz. teploty ($^{\circ}$ C)	–20 až 40
Výstupní napětí (mV/V FS)	0,8 \pm 0,1	Rozteč mont. otvorů H	6 mm
Chyba linearity (% FS)	\pm 0,03	Rozteč mont. otvorů V	40 mm
Chyba nuly (% FS)	\pm 0,05	Materiál	hliník
Teplotní drift (% FS/10 $^{\circ}$ C)	\pm 0,05	Rozměry (mm)	47 x 12 x 6

Rozměry a montáž



ZAPOJENÍ



00101
01001
00001

UKÁZKA PROGRAMU

```
// Arduino AD 24-bit převodník s 2 kanály

// piny pro připojení SCK a DT z modulu
int pSCK = 2;
int pDT = 3;

// definování různých nastavení kanálů a jejich zesílení
#define kanal_A_zesil_128 1
#define kanal_B_zesil_32 2
#define kanal_A_zesil_64 3
```

```

void setup ()
{
  // nastavení pinů modulu jako výstup a vstup
  pinMode (pSCK, OUTPUT);
  pinMode (pDT, INPUT);
  // komunikace přes sériovou linku rychlostí 9600 baud
  Serial.begin (9600);
  // probuzení modulu z power-down módu
  digitalWrite (pSCK, LOW);
  // spuštění prvního měření pro nastavení měřícího kanálu
  spusteniMereni (kanal_A_zesil_64);
}

void loop ()
{
  // výpis měření a jeho výsledku
  Serial.print ("Vysledek mereni: ");
  Serial.print (spusteniMereni (kanal_A_zesil_64));
  Serial.println (" uV");
  // pauza 0.5 s pro přehledné čtení
  delay (500);
}

// vytvoření funkce pro měření z nastaveného kanálu
long spusteniMereni (byte mericiMod)
{
  byte index;
  long vysledekMereni = 0L;
  // načtení 24-bit dat z modulu
  while (digitalRead (pDT));
  for (index = 0; index < 24; index++)
  {
    digitalWrite (pSCK, HIGH);
    vysledekMereni = (vysledekMereni << 1) | digitalRead (pDT);
    digitalWrite (pSCK, LOW);
  }
  // nastavení měřícího módu
  for (index = 0; index < mericiMod; index++)
  {
    digitalWrite (pSCK, HIGH);
    digitalWrite (pSCK, LOW);
  }
  // konverze z 24-bit dvojdoplňkového čísla
  // na 32-bit znaménkové číslo
  if (vysledekMereni >= 0x800000)
    vysledekMereni = vysledekMereni | 0xFF000000L;
  // přepočítání výsledku na mikrovolty podle zvoleného
  // kanálu a zesílení
  switch (mericiMod) {
    case 1: vysledekMereni = vysledekMereni / 128 / 2; break;
    case 2: vysledekMereni = vysledekMereni / 32 / 2; break;
    case 3: vysledekMereni = vysledekMereni / 64 / 2; break;
  }
  return vysledekMereni ;
}

```