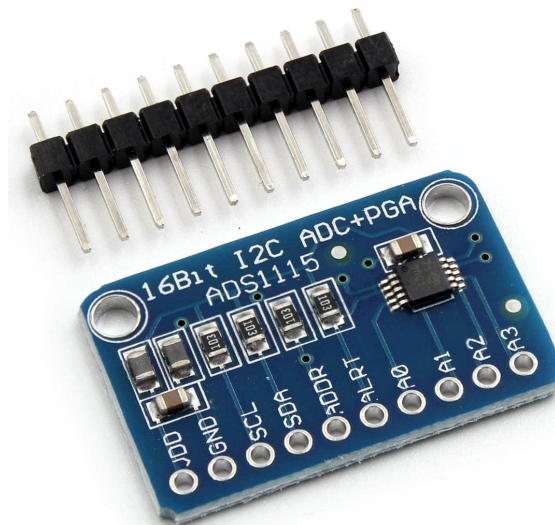


ADS1115 mit Arduino über I²C auslesen

ADS1115 4 Kanal AD-Wandler ADC Modul Breakout Board I2C Arduino Raspberry 16Bit

Versorge das Breakout für diesen Arduino-Uno-Test mit 5 V, verbinde ADDR mit GND und lege den Schleifer eines Potentiometers an A0. Der Code nutzt bewusst GAIN_TWOTHIRDS, damit ein Signal bis knapp 5 V nicht bereits am eingestellten Messbereich von $\pm 4,096$ V abschneidet.



Produkt bei Makershop:

<https://www.makershop.de/module/schnittstellen/ads1115-4-kanal-wandler/>

SKU 18443 | Stand 26.07.2026

Ein Potentiometer mit Arduino über I²C auslesen

Versorge das Breakout für diesen Arduino-Uno-Test mit 5 V, verbinde ADDR mit GND und lege den Schleifer eines Potentiometers an A0. Der Code nutzt bewusst GAIN_TWOTHIRDS, damit ein Signal bis knapp 5 V nicht bereits am eingestellten Messbereich von $\pm 4,096$ V abschneidet.

VDD	Arduino Uno 5 V
GND	Arduino Uno GND und äußerer Potentiometeranschluss
SDA	Arduino Uno A4 / SDA
SCL	Arduino Uno A5 / SCL
ADDR	GND für Adresse 0x48
A0	Schleifer eines Potentiometers zwischen 5 V und GND

Schritt für Schritt

1. Ausführung prüfen

Vergleiche Pinbeschriftung und sichtbare Bauteile des gelieferten Breakouts mit den Produktfotos. Unterstelle keinen Spannungsregler, Pegelwandler oder Eingangsschutz.

2. Stromlos verdrahten

Verbinde VDD, GND, SDA, SCL und ADDR bei ausgeschalteter Versorgung. Lege das Potentiometer zwischen 5 V und GND und seinen Schleifer an A0.

3. Bibliothek installieren

Installiere in der Arduino IDE die Bibliothek Adafruit ADS1X15; Adafruit BusIO wird als Abhängigkeit benötigt.

4. Adresse und Gain setzen

Starte den ADS1115 auf 0x48 und wähle GAIN_TWOTHIRDS für den Bereich $\pm 6,144$ V. Das erlaubt die Skalierung eines 0-bis-5-V-Signals, aber niemals eine Eingangsspannung oberhalb VDD.

5. Ersten Messwert sehen

Öffne den seriellen Monitor mit 115200 Baud. Beim Drehen des Potentiometers müssen Rohwert und Spannung gleichmäßig steigen oder fallen.

6. Messung plausibilisieren

Vergleiche die Anzeige an den Endstellungen mit GND und VDD. Bei springenden Werten Leitungen verkürzen, Masseführung prüfen und gegebenenfalls mehrere Messungen mitteln.

Beispiel

```

#include <Wire.h>
#include <Adafruit_ADS1X15.h>

Adafruit_ADS1115 ads;

void setup() {
  Serial.begin(115200);
  if (!ads.begin(0x48)) {
    Serial.println("ADS1115 nicht gefunden");
    while (true) delay(10);
  }

  // Für ein 0-bis-5-V-Signal: FSR ±6,144 V.
  // Die Eingangsspannung darf trotzdem nie über VDD liegen.
  ads.setGain(GAIN_TWOTHIRDS);
}

void loop() {
  int16_t raw = ads.readADC_SingleEnded(0);
  float volts = ads.computeVolts(raw);

  Serial.print("A0: ");
  Serial.print(raw);
  Serial.print(" | ");
  Serial.print(volts, 4);
  Serial.println(" V");
  delay(500);
}

```

Vertiefung

Projektziel und erstes Erfolgserlebnis

Ein Potentiometer an A0 drehen und im seriellen Monitor Rohwert sowie reale Spannung beobachten.

Boardvariante vor dem Start prüfen

Pinbeschriftung, Versorgung und sichtbare Bestückung kontrollieren; keine nicht bestätigten Schutz- oder Pegelwandlerfunktionen voraussetzen.

Arduino Uno sicher verdrahten

VDD an 5 V, gemeinsame Masse, SDA an A4, SCL an A5, ADDR an GND und Potentiometerschleifer an A0 anschließen.

Adafruit ADS1X15 installieren

Bibliothek und Abhängigkeit Adafruit BusIO über die Arduino-Bibliotheksverwaltung installieren.

Direkt nutzbarer Arduino-Code

ADS1115 auf Adresse 0x48 starten, GAIN_TWOTHIRDS setzen, A0 lesen und den Rohwert mit computeVolts in Volt umrechnen.

Gain sinnvoll auswählen

Messbereich an das erwartete Signal anpassen, ohne aus einer größeren PGA-Skalierung eine höhere zulässige Eingangsspannung abzuleiten.

Weitere Kanäle und Differenzmessung

A1 bis A3 ergänzen und differentielle Messungen nur innerhalb des zulässigen Gleichtakt- und Eingangsspannungsbereichs durchführen.

Fehlersuche und bessere Messwerte

Adresse scannen, gemeinsame Masse prüfen, Leitungen verkürzen, Datenrate reduzieren und bei Bedarf einen geeigneten RC-Filter oder Mittelwert einsetzen.

Höhere Spannungen messen

Spannungsteiler dimensionieren, Eingangsimpedanz und Schutzfälle berücksichtigen und 9 V oder 12 V niemals direkt an einen Analogeingang legen.

Quellen und nächste Schritte

TI-Datenblatt für Grenzwerte und Registerdetails sowie Adafruit-Dokumentation für Bibliotheksfunktionen und weiterführende Beispiele verwenden.

Häufige Fragen

Hat das Modul vier gleichzeitig arbeitende 16-Bit-ADCs?

Nein. Der ADS1115 besitzt einen ADC mit vorgeschaltetem Multiplexer. Die ausgewählten Kanäle werden nacheinander gewandelt; bei vier Kanälen verteilt sich die verfügbare Messrate entsprechend auf die Abfragen.

Darf A0 bei GAIN_TWOTHIRDS bis 6,144 V bekommen?

Nein. $\pm 6,144$ V beschreibt den internen Full-Scale-Bereich der Umrechnung. Laut TI dürfen die Analogeingänge selbst höchstens bis $VDD + 0,3$ V belastet werden; für einen robusten Aufbau bleiben sie zwischen GND und VDD.

Kann ich eine 9-V- oder 12-V-Batterie direkt messen?

Nein. Spannungen oberhalb der Modulversorgung benötigen mindestens einen passend berechneten Spannungsteiler und eine zum Projekt passende Schutzbeschaltung. Auch Fehlerfälle und die Belastbarkeit der Widerstände müssen berücksichtigt werden.

Kann der ADS1115 negative Spannungen messen?

Das Ergebnis einer Differenzmessung kann negativ sein, wenn der negative Eingang das höhere Potenzial hat. Keiner der beiden Eingangspins darf dabei außerhalb der zulässigen Spannung relativ zu GND und VDD liegen.

Warum stehen bei einer single-ended Messung nicht alle 16 positiven Bits zur Verfügung?

Der ADS1115 liefert sein 16-Bit-Ergebnis im Zweierkomplement. Single-ended gegen GND nutzt nur die positive Hälfte des Wertebereichs von 0x0000 bis 0x7FFF; kleine negative Codes nahe null können durch Offset entstehen.

Kann ich mehrere ADS1115 an einem I²C-Bus betreiben?

Ja. ADDR erlaubt die vier Adressen 0x48 bis 0x4B, sodass bis zu vier Bausteine am selben Bus möglich sind. Bei mehreren Modulen müssen vorhandene Pull-ups, Buskapazität und Leitungslänge zusammen betrachtet werden.

Warum schwankt der Messwert trotz 16 Bit?

Auflösung ist nicht dasselbe wie absolute Genauigkeit. Rauschen, Masseführung, Quellimpedanz, Datenrate, PGA-Einstellung und Versorgung beeinflussen das Ergebnis. Kurze Leitungen, ein geeigneter RC-Filter und Mittelung können helfen.

Ist das Modul identisch mit dem aktuellen Adafruit ADS1115 Breakout?

Nein, das wird nicht behauptet. Adafruit-Dokumentation und -Bibliothek sind für Chipfunktion und Software hilfreich; Anschlüsse und Zusatzbeschaltung des angebotenen unbranded Breakouts richten sich ausschließlich nach der gelieferten Variante.

Verlässliche Quellen

Texas Instruments ADS1113, ADS1114, ADS1115 Datasheet (Rev. E)
<https://www.ti.com/lit/ds/symlink/ads1115.pdf>

Adafruit 4-Channel ADC Breakouts – Pinouts und Signalanschlüsse
<https://learn.adafruit.com/adafruit-4-channel-adc-breakouts/pinouts>

Adafruit 4-Channel ADC Breakouts – Arduino Code
<https://learn.adafruit.com/adafruit-4-channel-adc-breakouts/arduino-code>

Adafruit ADS1X15 Arduino Library
https://github.com/adafruit/Adafruit_ADS1X15

Mehr Wissen für dein Projekt

Aktuelle Produktinformationen, weitere Dokumente und passende Komponenten findest du direkt bei Makershop.

<https://www.makershop.de/module/schnittstellen/ads1115-4-kanal-wandler/>