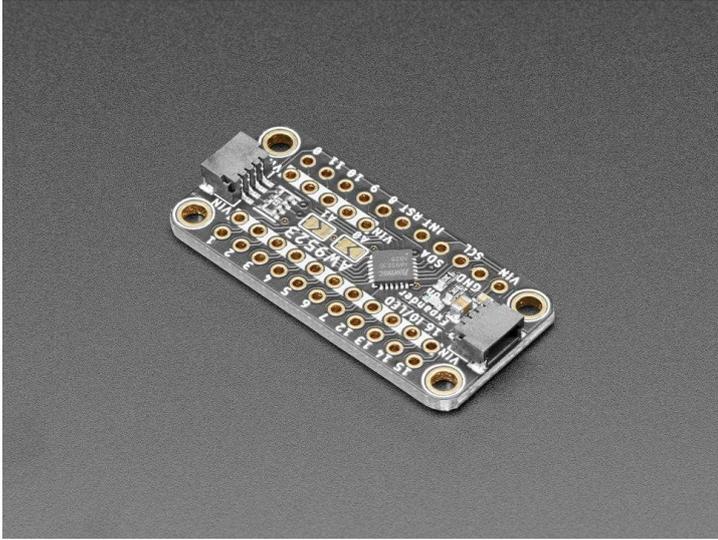




Adafruit AW9523 GPIO Expander und LED Treiber Breakout



Numero Ordine:	ADA4886
Hersteller:	Adafruit
Herkunftsland:	USA
Zolltarifnummer:	85423111
Gewicht:	0.005 kg

Erweitern Sie Ihre Projektmöglichkeiten mit dem **Adafruit AW9523 GPIO Expander und LED Treiber Breakout** - ein niedlicher und leistungsstarker I2C Expander mit einer Menge Tricks im Ärmel.

GPIO-Expander funktionieren so: Sie haben ein Board mit einer gewissen Anzahl von GPIO, aber nicht genug für Ihr Projekt - vielleicht brauchen Sie mehr Tasten oder LEDs. Sie *könnten* auf ein Board mit einer massiven Anzahl von GPIOs aufrüsten, wie das Grand Central, oder Sie könnten eines dieser Boards aufstecken. Schließen Sie es über I2C an und dann können Sie I2C-Befehle senden/empfangen, um die GPIO-Pins zu steuern, um sie zu schreiben und zu lesen. Das wird zwar langsamer sein als der direkte GPIO-Zugriff, aber vielleicht macht das nichts, wenn es eine Millisekunde statt einer Mikrosekunde dauert. Man braucht nur die beiden I2C-Pins und kann den I2C-Port sogar mit anderen Sensoren und Geräten teilen. Sie können sogar weitere Expander für eine massive I/O-Steuerung hinzufügen!

Der AW9523 ist eine Abwandlung des üblichen I2C-Expanders:

- Erstens ist er sehr günstig - wer liebt das nicht?
- Es hat 16 I/O-Pins, das verdoppelt die Pinanzahl der meisten Boards
- Vier I2C-Adressoptionen, so dass Sie 4 Expander an einen Bus anschließen können
- Jeder Pin kann ein Eingang oder ein Ausgang sein
- IRQ-Ausgang kann Sie alarmieren, wenn die Eingangspins ihren Wert ändern
- Dieser Chip *unterstützt keine internen Pull-ups oder Pull-downs*, Sie müssen einen externen Widerstand hinzufügen, wenn Sie einen benötigen
- Der Chip *unterstützt* jedoch 8-bit lineare Konstantstrom-LED-Dimmung, so dass Sie LEDs ohne Widerstände anschließen können und eine großartig aussehende Dimmung ohne PWM haben
- Die ersten 8 Pins können als Open Drain konfiguriert werden (als Gruppe)

Das Fehlen von intern konfigurierbaren Pulls ist ein bisschen schade, aber wir denken, dass der Expander dies durch die Konstantstrom-LED-Ansteuerung mehr als wettmacht. Wenn Sie einen Expander verwenden, um viele steuerbare LEDs hinzuzufügen, wird dieses Board es sehr einfach machen. Da es sich um eine Konstantstrom-LED handelt, brauchen Sie *keine Widerstände* in der Reihe mit jeder LED (obwohl es nicht schadet, wenn Sie es tun): Schließen Sie einfach die LED-Anode an eines der vielen VIN-Pads an und verbinden Sie dann die Kathode mit dem GPIO-Pin.

Natürlich können Sie mit den Pins auch beliebige Taster oder andere I/Os ansteuern - wir finden nur, dass sich dieses Board besonders gut für die Ansteuerung von LEDs eignet. Es gibt auch einen Interrupt-Ausgang, Sie können den Pin-Änderungs-IRQ für beliebige Pins aktivieren, so dass Sie benachrichtigt werden können, wenn es Zeit ist, die I/O-Zustände zu lesen.

Eine Besonderheit dieses Chips ist, dass die Standard-I2C-Adresse den anfänglichen Boot-Zustand der Pins bestimmt. Unsere Bibliotheken führen sofort einen Soft-Reset durch und konfigurieren alle Pins auf Eingänge und Push-Pull, so dass Sie das gleiche Verhalten erwarten können, egal wie die I2C-Adresse lautet. Wir empfehlen Ihnen jedoch, das Datenblatt Tabelle 1 zu prüfen, um



sicherzustellen, dass Ihre Hardware davon nicht betroffen ist.

Wir haben sowohl [Arduino-](#) als auch [CircuitPython/Python-Bibliotheken](#) für die AW9523 geschrieben, so dass Sie sofort loslegen können, egal ob Sie ein Arduino-kompatibles UNO oder ein Raspberry Pi 4 haben - oder irgendetwas dazwischen.

Damit Sie schnell loslegen können, haben wir eine speziell angefertigte Platine imSTEMMA QT Formfaktor entwickelt, die einfach zu bedienen ist. Die [STEMMA QT-Anschlüsse](#) auf beiden Seiten sind kompatibel mit den SparkFun Qwiic I2C-Anschlüssen. Damit können Sie lötfreie Verbindungen zwischen Ihrem Entwicklungsboard und dem AW9523 herstellen oder es mit einer Vielzahl anderer Sensoren und Zubehöerteilen über ein kompatibles Kabel verbinden.

QT-Kabel ist nicht im Lieferumfang enthalten.

Weitere Bilder:

