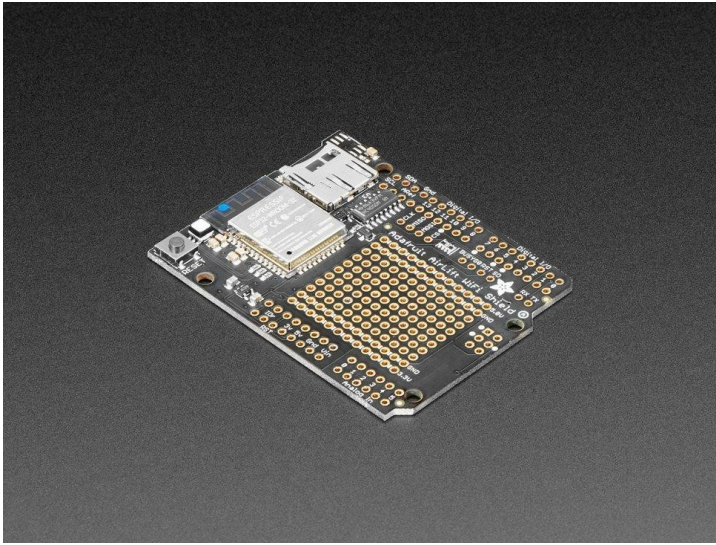




## Adafruit AirLift Shield - ESP32 WLAN Co-Prozessor



|                  |          |
|------------------|----------|
| Artikel-Nr.:     | ADA4285  |
| Hersteller:      | Adafruit |
| Herkunftsland:   | USA      |
| Zolltarifnummer: | 84718010 |
| Gewicht:         | 0.02 kg  |

Mit dem Adafruit AirLift Shield können Sie den leistungsstarken ESP32 als WLAN-Coprozessor verwenden. Wahrscheinlich haben Sie Ihren bevorzugten Arduino-kompatiblen wie den Metro M4 oder den klassischen Metro 328), der mit seiner eigenen Reihe von großartigen Peripheriegeräten und vielen Bibliotheken kommt. Aber er hat kein WLAN eingebaut! Also geben wir diesem Chip einen besten Freund, den ESP32. Dieser Chip übernimmt alle schweren Aufgaben der Verbindung mit einem WLAN-Netzwerk und die Übertragung von Daten von einer Website, auch wenn es mit der neuesten TLS/SSL-Verschlüsselung (es hat Root-Zertifikate vorgebrannt).

Da WLAN von einem separaten Chip verwaltet wird, ist Ihr Code einfacher, Sie müssen keine Socket-Daten zwischenspeichern oder eine SSL-Bibliothek einkompilieren und debuggen. Senden Sie einfache, aber leistungsfähige Socket-basierte Befehle über 8MHz SPI für Hochgeschwindigkeits-Datentransfer. Sie können jeden 3V- oder 5V-Arduino verwenden, jeden Chip ab dem ATmega328 oder aufwärts (obwohl der '328 nicht in der Lage sein wird, sehr komplexe Aufgaben zu erledigen oder eine Menge Daten zu puffern). Es funktioniert auch sehr gut mit CircuitPython, ein SAMD51/Cortex M4 ist mindestens erforderlich, da wir eine Menge RAM benötigen. Alles, was man braucht, ist der SPI-Bus und 2 Steuerepins plus eine Stromversorgung, die bei WLAN-Nutzung bis zu 250mA liefern kann.

Wir haben ein ESP32-Modul auf ein Shield mit einem separaten 3,3-V-Regler und einem Tri-State-Chip für MOSI gesetzt, damit man den SPI-Bus mit anderen Shields teilen kann. Wir haben auch einen Micro-SD-Kartensockel aufgesetzt, den man zum Hosten oder Speichern von Daten, die man aus dem Internet bekommt, verwenden kann. Arduinos, die auf dem ATmega328 basieren (wie der UNO), können nicht gleichzeitig das WiFi-Modul und die SD-Bibliothek nutzen, da sie nicht genug RAM haben. Auch hier empfehlen wir einen M0 oder M4 Chipsatz für die Verwendung mit Arduino, M4 für CircuitPython!

Kommt komplett zusammengebaut und getestet, vorprogrammiert mit ESP32 SPI WiFi Co-Prozessor-Firmware, die [Sie in CircuitPython verwenden können](#). Wir legen auch einige Header bei, so dass Sie es einlöten und direkt in Ihr Arduino-kompatibles Board einstecken können, aber Sie können auch einen Satz Stacking-Header nehmen, um über/unter Ihrem Board zu stapeln.

Wir haben dies mit all unseren Metros getestet und es sollte mit allen funktionieren, außer mit den Metro M4 Airlifts (denn die haben bereits WLAN!).

**Für den Einsatz in Arduino**, die '328 und '32u4 beherrschen grundlegende Konnektivität und Datentransfer, aber sie haben nicht viel RAM, so dass wir sie nicht empfehlen - verwenden Sie die Metro M0, M4 oder ähnliche, für beste Ergebnisse!

**Für den Einsatz von CircuitPython** funktioniert eine Metro M4 am besten - die M0-Serie hat nicht genug RAM in CircuitPython.

Die [Firmware](#) auf dem Board ist eine leichte Variante des Arduino WiFinINA-Kerns, die hervorragend funktioniert!

Schauen Sie sich die Anleitung für das [Learning System](#) an, um Schaltpläne und Dateien zu erhalten und innerhalb weniger Minuten mit AirLift'in' loszulegen!



## Weitere Bilder:

