

AZ-Delivery

Willkommen!

Vielen Dank, dass sie sich für unser *Bewegungserkennungs Set* von AZ-Delivery entschieden haben. In den nachfolgenden Seiten werden wir Ihnen erklären wie Sie das Gerät einrichten und nutzen können.

Viel Spaß!



Az-Delivery

Inhaltsverzeichnis

Einführung.....	3
Einführung in das ESP32 Dev Kit C	4
Technische Daten des ESP32 Dev Kit C	5
Pinbelegung des ESP32 Dev Kit C.....	6
Einführung in das PIR-Sensor-Modul.....	7
Technische Daten des PIR Sensors.....	8
Pinbelegung des PIR Sensors.....	10
Einrichtung der Arduino IDE.....	11
Verbindung des PIR Sensors mit dem ESP32 Dev Kit C.....	18

Einführung

Das Bewegungserkennung Set umfasst das ESP32 Dev Kit C, den HC-SR501 PIR-Sensor und ein Mini-Breadboard. Das ESP32 Dev Kit C wird als Hauptmikrocontroller verwendet, der HC-SR501 PIR-Sensor als Bewegungserkennungssensor und ein Breadboard als Verbindung zwischen ihnen.

Das ESP32 Dev Kit C arbeitet im 3,3V-Bereich, der HC-SR501 PIR-Sensor arbeitet jedoch im 5V-Bereich, was bedeutet, dass er für den Betrieb mit dem ESP32 Dev Kit C modifiziert werden muss. Standardmäßig verfügt der PIR-Sensor HC-SR501 über einen integrierten 5V-Spannungsregler. Um den Sensor mit der 3,3V-Stromversorgung zu versorgen, muss der On-Board-Spannungsregler überbrückt werden. Dieses Verfahren wird im Kapitel "Wie wird der PIR-Sensor mit der 3,3V-Stromversorgung verwendet?", erklärt.

Einführung in das ESP32 Dev Kit C

Das ESP32 Dev Kit C ist ein Entwicklungs-Board, das um den *ESP32*-Chip herum entwickelt wurde. Außer dem Chip enthält das Board auch einen 3,3V-Spannungsregler, einen USB-Programmierschaltkreis für den *ESP32*-Chip und einige andere Funktionen.

Das ESP32 Dev Kit C wird mit einer vorinstallierten Firmware geliefert, die es ermöglicht, das Board mit der interpretierten LUA-Language zu programmieren und Befehle über die serielle Schnittstelle (CP2102-Chip) zu senden. Das ESP32 Dev Kit C-Board ist eine der am häufigsten verwendeten Plattformen für Internet of Things (IoT)-Projekte. Das Board ist vollständig kompatibel mit der Arduino IDE, Espressif IDF (IoT Development Framework), microPython, JavaScript usw.

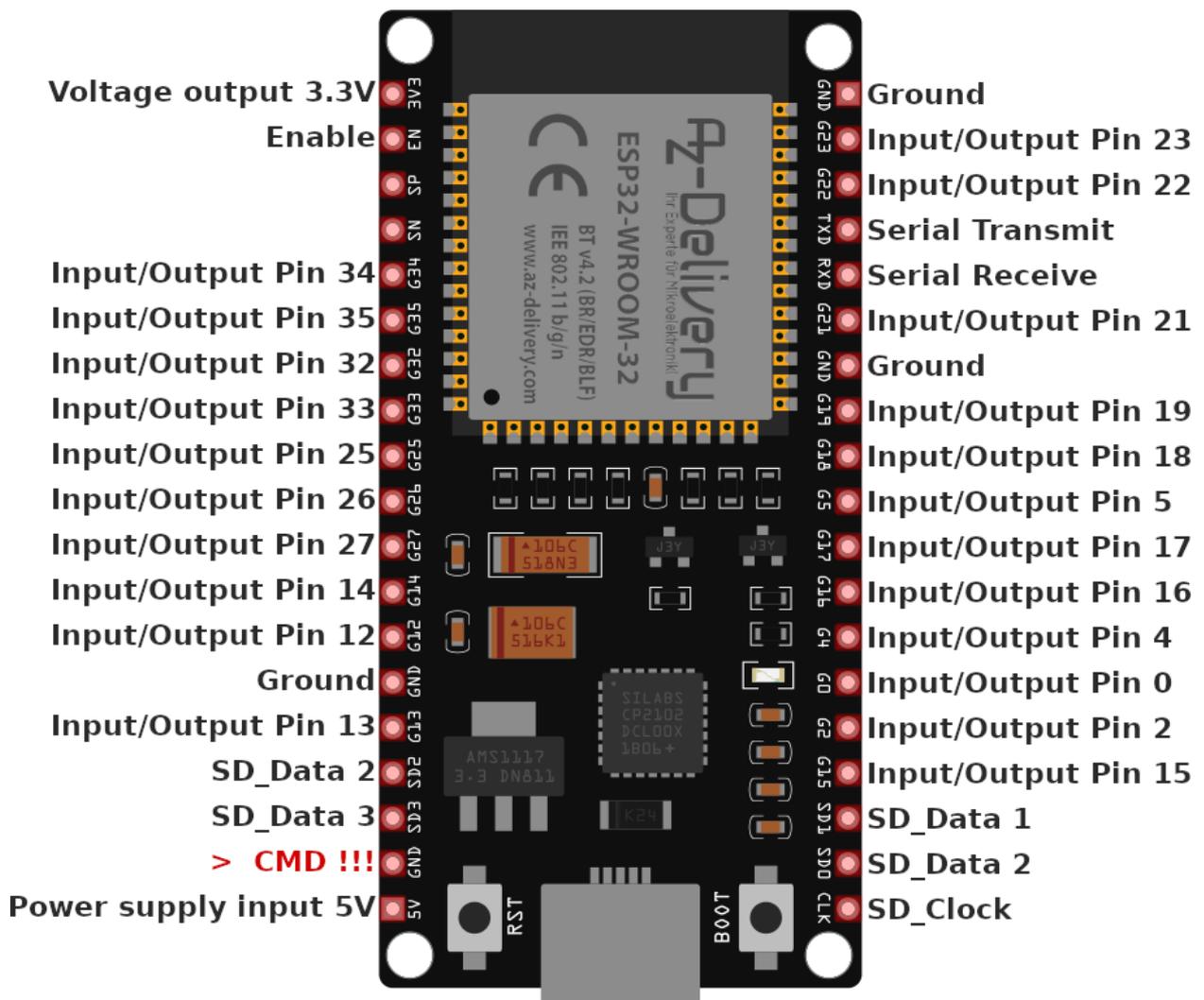
Das ESP32 Dev Kit C-Board ist speziell für die Arbeit mit dem Breadboard konzipiert. Es verfügt über einen 3,3V-Spannungsregler, der die Stromversorgung direkt über den USB-Port ermöglicht. Die Eingangs-/Ausgangspins arbeiten mit 3,3V. Der On-Board-Chip *CP2102* wird als USB-zu-serielle Schnittstelle verwendet.

Technische Daten des ESP32 Dev Kit C

Stromversorgungsspannung (USB)	5V
Eingangs-/Ausgangsspannung	3.3V
SoC	ESP32-WROOM-32
CPU	Xtensa 32-bit LX6 Mikroprozessor
Taktfrequenz	Bis zu 240MHz
Anweisungs-RAM	512kB
Externer Flash-Speicher	Bis zu 4MB
Digitale GPIO-Pins	16 (auch als PWM konfigurierbar)
Analog-Digital-Pins	18
Analog-Digital Auflösung	12bit
Digital-Analog-Pins	2
Digital-Analog Auflösung	8bit
UART	3
I2C-Schnittstellen	2
I2S-Schnittstellen	2
SPI-Schnittstellen	4
Berührungssensor-Eingänge	10
IR-Fernbedienungs-Pins	8
Integrierter Halleffekt-Sensor	
Analoger Vorverstärker (Low Power)	
Mehrere Echtzeituhren	
USB-Seriell-Chip	CP2102
PCB-Antenne	
Leistung des Antennenausgangs	+ 19.5dBm in 802.11b Modus
Wi-Fi	2.4 GHz bis zu 150 Mbits/s
Drahtlose Protokolle	802.11 b/g/n
Bluetooth	BLE und legacy Bluetooth
Integrierter TCP/IP-Protokollstapel	

Pinbelegung des ESP32 Dev Kit C

Das ESP32 Dev Kit C hat 38 Pins. Pinbelegung wird unten abgebildet:



Einführung in das PIR-Sensor-Modul

Das PIR-Sensormodul HC-SR501 ist ein Gerät, das Veränderungen im Infrarotlicht erkennen kann. Beispielsweise sendet der menschliche Körper mit einer Temperatur von 36°C (97°F) infrarotes Licht aus, das erkannt werden kann, wenn es sich im Abtastbereich des Moduls befindet. Der PIR-Sensor ist ein hochempfindlicher, hochzuverlässiger Sensor und arbeitet mit extrem niedriger Spannung. Das Modul basiert auf Infrarot-Technologie und verwendet einen pyroelektrischen passiven Infrarotsensor, kurz PIR. Die *Fresnel*-Linse wird verwendet, um so viel Umgebungslicht wie möglich auf den Erfassungsbereich des Sensors zu fokussieren, wodurch der Erfassungswinkelbereich vergrößert wird.

Wenn das kristalline Element im Sensor dem Infrarotlicht ausgesetzt wird, wird die Ausgangsspannung direkt mit der Intensität des Infrarotlichts verändert. Die Ausgangsspannung des Sensors liegt im Bereich von mV. Diese Spannung wird über den On-Board-Verstärker verstärkt, so dass der Mikrocontroller sie erkennen kann.

Das Modul kann mit verschiedenen Arten von Geräten wie Lichtgeräten, Gegensprechanlagen, elektrischen Ventilatoren und anderen Haushaltsgeräten verwendet werden. Es wird oft in Häusern, Unternehmen, Hotels, Geschäften, sensiblen Bereichen, in denen Lichtautomatisierungs- und Sicherheitssysteme benötigt werden, verwendet.

Technische Daten des PIR Sensors

Stromverorgunsspannung:	5V DC
Betriebstemperatur:	-15 bis +70°C
Sperrzeit:	200 Milisekunden
Empfindlichkeit:	110 Grad
Abstandsbereich:	Bis zu 7 meters
Blockierzeit:	2.5s. (Standart) bis zu 10 s.
Auslösende Methoden:	L - deaktivieren, H – Wiederholungstrigger aktivieren
TTL-Ausgang:	LOW – 0V, HIGH – 3.3V (Logik-Level)
Ruhestrom:	Weniger als 50µA
Stromverbrauch:	65mA
Durchmesser der Linse:	23mm
Dimensionen:	33 x 25 x 30mm [1.3 x 1 x 1.2in]

Die Verzögerungszeit bestimmt, wie lange der PIR den Ausgang HIGH hält, nachdem eine Bewegung erkannt wurde.

Die Blockierzeit ist das Zeitintervall, in dem der Sensor deaktiviert ist oder keine Bewegung erkennt. Die Standard-Blockierzeit für den Sensor beträgt 2,5 Sekunden.

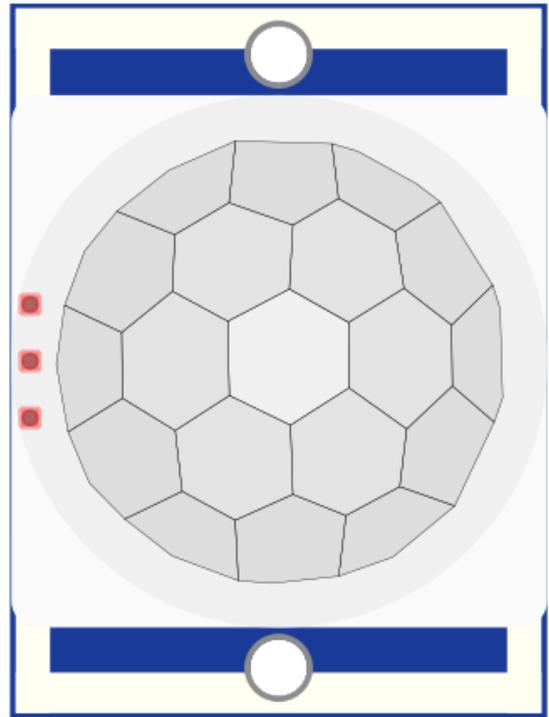
Die Sperrzeit stellt das Zeitintervall dar, in dem der Sensor das Objekt in dem Bereich erkennt.

Die Auslösemethode ist eine über einen Jumper wählbare Funktion, die Teil der Temperaturkompensationseinstellungen ist.

Pinbelegung des PIR Sensors

Das HC-SR501 PIR Modul hat drei Pins. Die Pinbelegung wird unten abgebildet:

GROUND - GND
DIGITAL OUTPUT - OUT
POWER SUPPLY - VCC



Az-Delivery

Einrichtung der Arduino IDE

Falls die Arduino-IDE nicht installiert ist, klicken Sie auf folgenden [link](#) und laden Sie die Installationsdatei für das Betriebssystem Ihrer Wahl herunter.

Download the Arduino IDE



The screenshot shows the Arduino IDE download page. On the left, there is a teal circle containing the Arduino infinity logo. To its right, the text reads: **ARDUINO 1.8.9**. Below this, it states: "The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software. This software can be used with any Arduino board. Refer to the [Getting Started](#) page for Installation instructions." On the right side of the page, there are several download options: "Windows Installer, for Windows XP and up", "Windows ZIP file for non admin install", "Windows app Requires Win 8.1 or 10" with a "Get" button, "Mac OS X 10.8 Mountain Lion or newer", "Linux 32 bits", "Linux 64 bits", "Linux ARM 32 bits", and "Linux ARM 64 bits". At the bottom right, there are links for "Release Notes", "Source Code", and "Checksums (sha512)".

Für *windows*-User: Doppelklicken Sie auf die heruntergeladene *.exe*-Datei und folgen Sie den Anweisungen im Installationsfenster.

Az-Delivery

Für Linux laden Sie die Datei mit der Erweiterung ".tar.xz" herunter, die Sie dann extrahieren müssen. Wenn Sie sie extrahieren, gehen Sie in das extrahierte Verzeichnis und öffnen Sie das Terminal in diesem Verzeichnis. Sie müssen zwei ".sh"-Skripte ausführen, das erste heißt "*arduino-linux-setup.sh*", das zweite "*install.sh*".

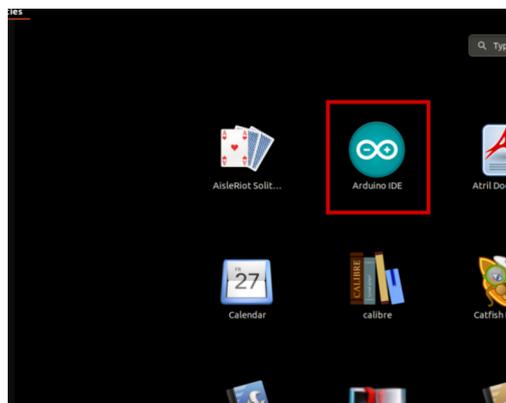
Um das erste Skript im Terminal (im extrahierten Verzeichnis) auszuführen, führen Sie den folgenden Befehl aus:

```
sh arduino-linux-setup.sh user_name
```

user_name - ist der Name des Superusers im Linux-Betriebssystem. Danach werden Sie aufgefordert, das Passwort für den Superuser einzugeben. Warten Sie einige Minuten, bis das Skript alles abgeschlossen hat.

Nach der Installation des ersten Skripts führen Sie das zweite Skript mit dem Namen "*install.sh*" aus. Führen Sie im Terminal (im extrahierten Verzeichnis) den folgenden Befehl aus: **sh install.sh**

Nach der Installation dieser Skripte, gehen Sie zu *All Apps*, um die installierte *Arduino IDE* zu finden.



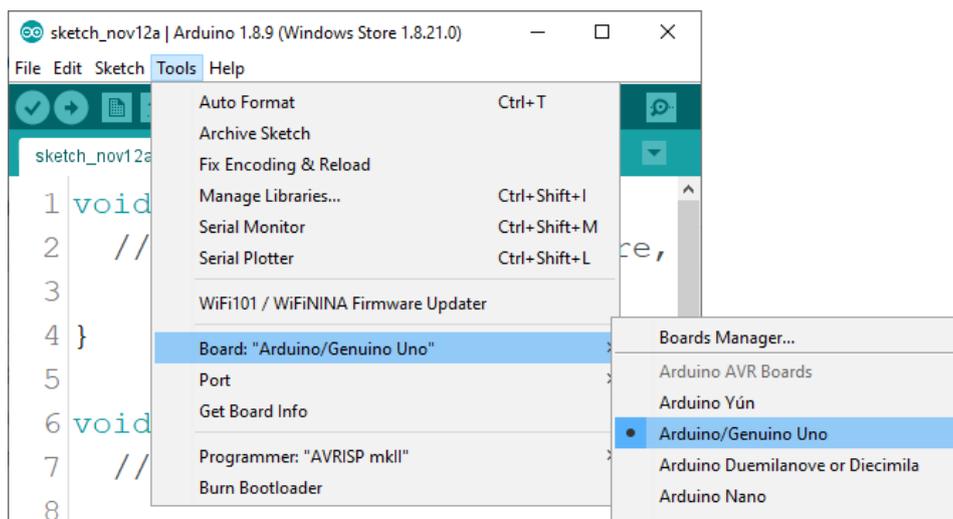
Az-Delivery

Fast alle Betriebssysteme werden mit einem vorinstallierten Texteditor ausgeliefert (z.B. Windows mit Notepad, Linux Ubuntu mit Gedit, Linux Raspbian mit Leafpad usw.). Alle diese Texteditoren sind für den Zweck des eBooks vollkommen in Ordnung.

Als nächstes sollten Sie überprüfen, ob Ihr PC das Uno board erkennen kann. Öffnen Sie die frisch installierte *Arduino IDE* und gehen Sie zu:

Tools > Board > {your board name here}

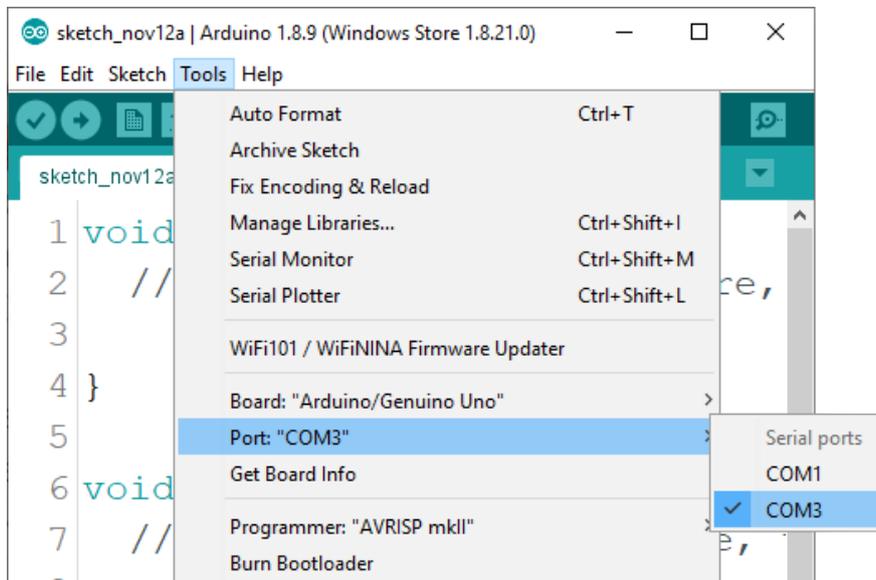
{your board name here} sollte das *Arduino/Genuino Uno* sein, wie Sie dem Bild unten entnehmen können:



Der Port, an den das Arduino-Board angeschlossen ist, muss ausgewählt werden. Gehe zu: *Tools > Port > {port name goes here}* und wenn das Arduino-Board an den USB-Port angeschlossen ist, ist der Portname im Drop-down Menü wie auf dem vorherigen Bild zu sehen.

Az-Delivery

Wenn die Arduino-IDE unter Windows verwendet wird, lauten die Portnamen wie ifolgt:



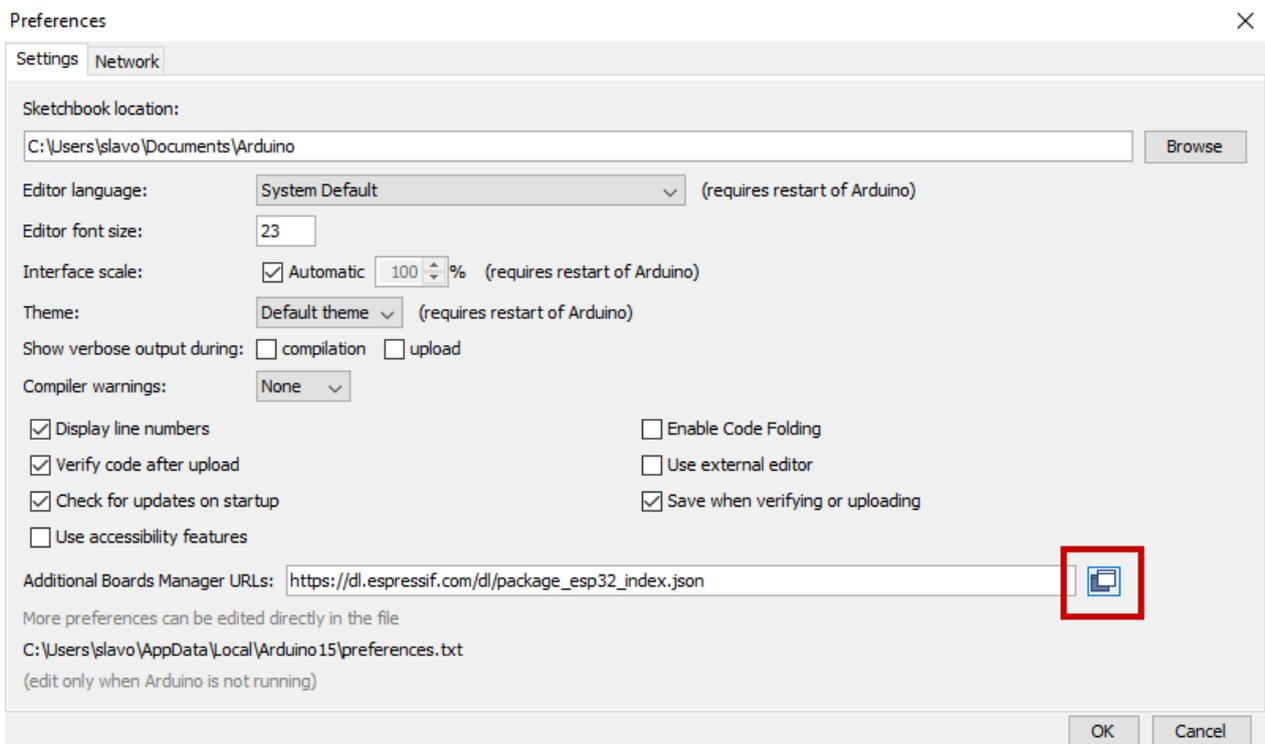
Für *Linux* Benutzer, ist zum Beispiel der Portname `/dev/ttyUSBx`, wobei x für eine ganze Zahl zwischen 0 und 9 steht.

Az-Delivery

Einrichtung der Arduino IDE für das ESP32 Dev Kit C

Um die Dev Kit C mit dem Arduino IDE zu verwenden, müssen Sie noch weitere Einstellungen in der Arduino IDE vornehmen.

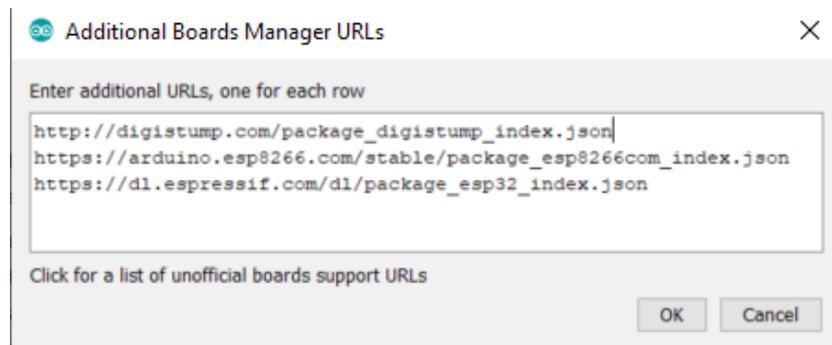
Als Erstes muss der ESP32 Kern installiert werden. Um ihn zu installieren, öffnen Sie die Arduino IDE und gehen Sie zu: *File > Preferences*



Az-Delivery

Fügen Sie in das *URL*-Feld des *Additional Boards Manager* Folgendes ein:
https://dl.espressif.com/dl/package_esp32_index.json

Wenn sich in diesem Feld ein oder mehrere Links befinden, drücken Sie auf die Schaltfläche im roten Rechteck (Bild auf der vorhergehenden Seite) und das folgende Fenster öffnet sich:



Fügen Sie nach dem letzten Link ein Komma ein und fügen Sie den Link von der vorherigen Seite nach dem Komma ein und klicken Sie auf die Schaltfläche *OK*. Klicken Sie dann im Fenster "Einstellungen" auf *OK*.

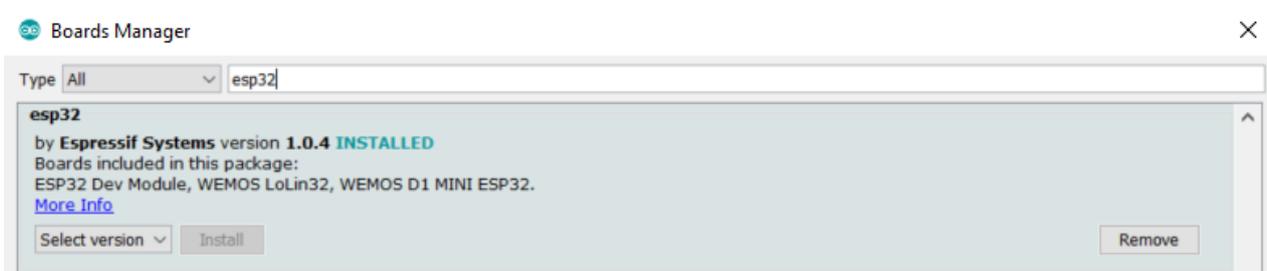
Az-Delivery

Dann müssen sie ESP32 Boards in der Arduino IDE installieren.

Gehen Sie zu:

Tools > Board > Boards Manager

Ein neues Fenster öffnet sich. Geben Sie "esp32" in das Suchfeld ein und installieren Sie das Board mit dem Namen "esp32" von "Espressif Systems", wie unten abgebildet:



Um das ESP32 Dev Kit C Board zu programmieren, muss das Board in der Arduino IDE ausgewählt werden. Gehen Sie zu:

Tools > Board > {board name}

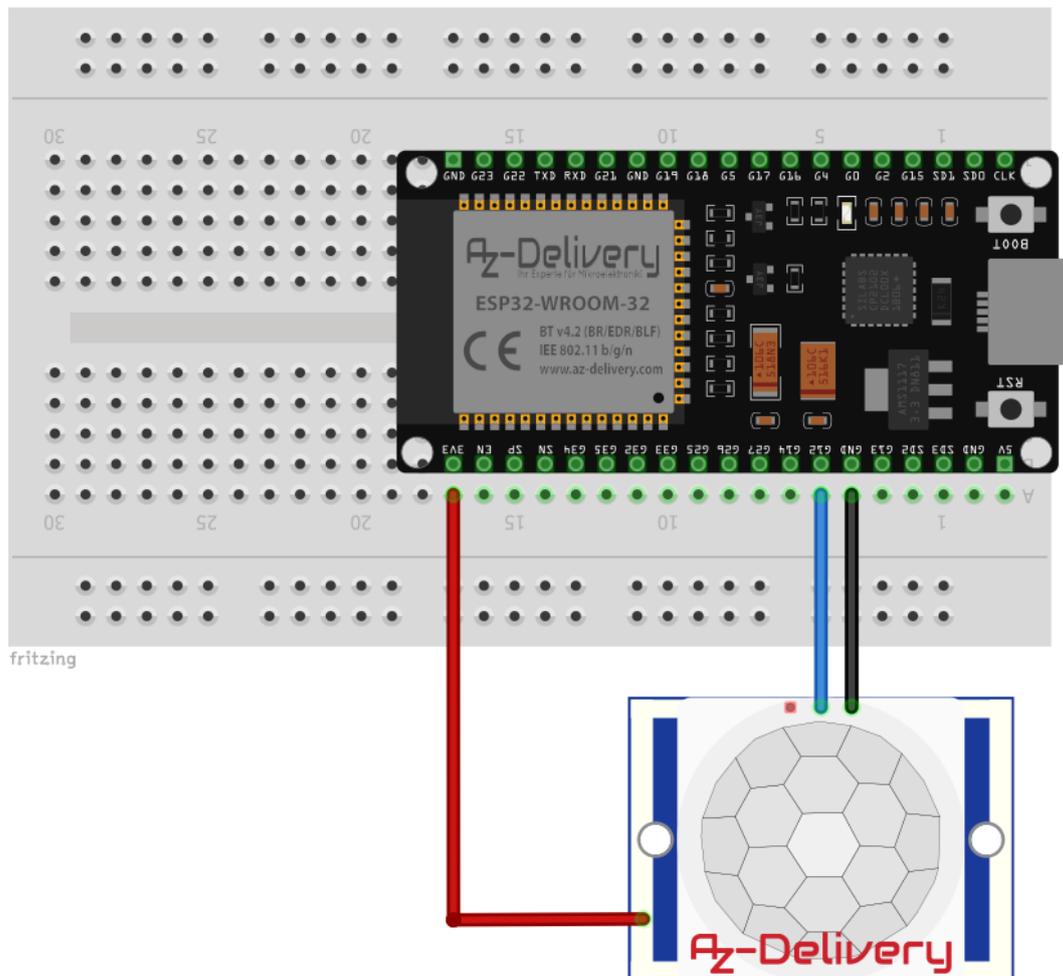
Wählen Sie "ESP32 Dev Module" aus

Um Sketche auf das ESP32 Dev Kit C Board hochzuladen, muss als nächstes der USB-Port, an den das Board angeschlossen ist, in der Arduino-IDE ausgewählt werden. Gehen Sie dafür zu:

Tools > Port > {port name}

Verbindung des PIR Sensors mit dem ESP32 Dev Kit C

Verbinden Sie den PIR Sensor mit dem ESP32 Dev Kit C wie unten abgebildet:



PIR Pin	>	ESP32 Dev Kit C Pin	
OUT	>	G12	Blauer Draht
GND	>	GND	Schwarzer Draht
H pin	>	3V3	Roter Draht

Az-Delivery

Sketch-Beispiel

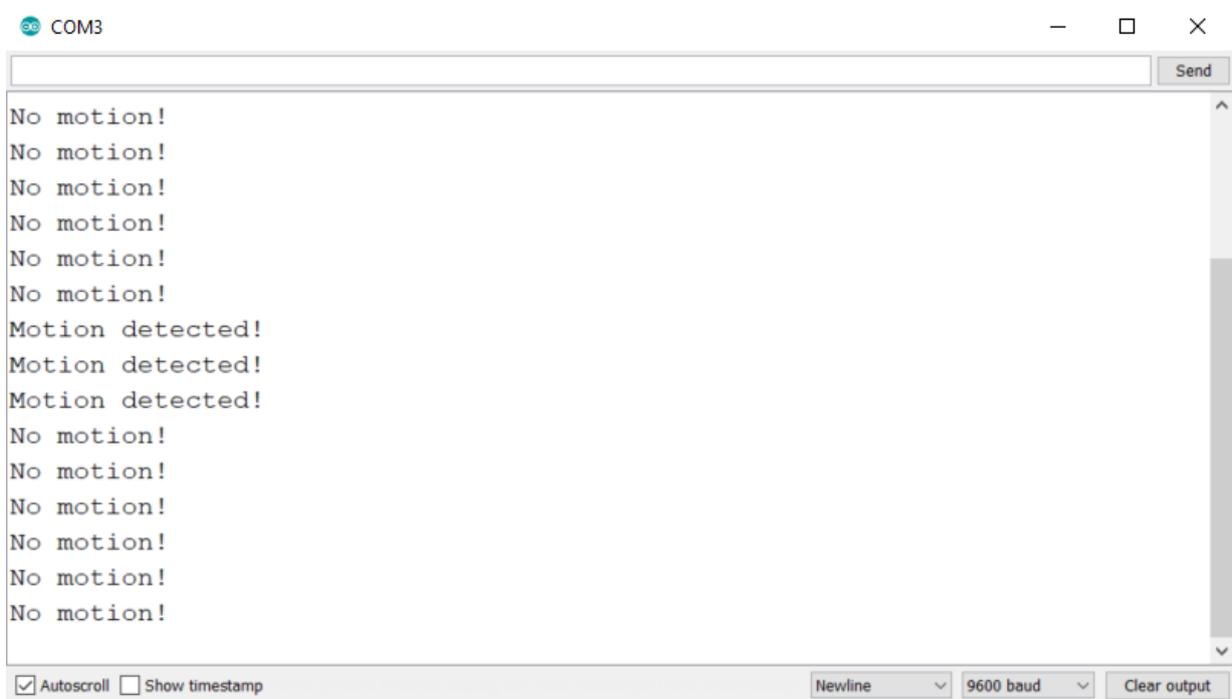
```
#define SENSOR_PIN 12

void setup() {
  Serial.begin(9600);
  pinMode(SENSOR_PIN, INPUT);
}

void loop(){
  boolean state = digitalRead(SENSOR_PIN);
  delay(1000);
  if(state == HIGH) {
    Serial.println("Motion detected!");
  }
  else {
    Serial.println("No motion!");
  }
}
```

Az-Delivery

Laden Sie den Sketch auf das ESP32-Dev Kit C und führen Sie den Serial Monitor aus (*Tools > Serial Monitor*). Die Ausgabe sollte wie folgt aussehen:



The screenshot shows the Serial Monitor window for COM3. The output text is as follows:

```
No motion!  
No motion!  
No motion!  
No motion!  
No motion!  
No motion!  
Motion detected!  
Motion detected!  
Motion detected!  
No motion!  
No motion!  
No motion!  
No motion!  
No motion!  
No motion!
```

At the bottom of the window, there are controls: Autoscroll, Show timestamp, a dropdown menu set to 'Newline', a dropdown menu set to '9600 baud', and a 'Clear output' button.

Sie haben es geschafft. Sie können jetzt unser Modul für Ihre Projekte nutzen.

AZ-Delivery

Jetzt sind Sie dran! Entwickeln Sie Ihre eigenen Projekte und Smart-Home Installationen. Wie Sie das bewerkstelligen können, zeigen wir Ihnen unkompliziert und verständlich auf unserem Blog. Dort bieten wir Ihnen Beispielskripte und Tutorials mit interessanten kleinen Projekten an, um schnell in die Welt der Mikroelektronik einzusteigen. Zusätzlich bietet Ihnen auch das Internet unzählige Möglichkeiten, um sich in Sachen Mikroelektronik weiterzubilden.

Falls Sie nach weiteren hochwertigen Produkten für Arduino und Raspberry Pi suchen, sind Sie bei AZ-Delivery Vertriebs GmbH goldrichtig. Wir bieten Ihnen zahlreiche Anwendungsbeispiele, ausführliche Installationsanleitungen, E-Books, Bibliotheken und natürlich die Unterstützung unserer technischen Experten.

<https://az-delivery.de>

Viel Spaß!

Impressum

<https://az-delivery.de/pages/about-us>