

what do i connect on the esp32s3 wroom

ChatGPT said:

You're trying to connect an **I2C OLED display** (based on the image) to an **ESP32-S3 WROOM**, and the image currently shows it wired to an **Arduino Uno**. Here's how to translate the connections from Arduino Uno to the **ESP32-S3 WROOM**.

From the Image (Arduino Uno to OLED):

- GND → GND
- VCC \rightarrow 5V
- SDA \rightarrow A4
- $SCL \rightarrow A5$

SP32-S3 WROOM Equivalent Wiring:

Important: Use 3.3V for VCC — most ESP32 boards are **not 5V-tolerant** on their I/O pins.

OLED Pin	ESP32-S3 Pin	Notes
GND	GND	Ground
VCC	3.3V	Do not use 5V , use 3.3V
SDA	GPIO 8 (or any I2C-capable pin)	Default SDA pin
SCL	GPIO 9 (or any I2C-capable pin)	Default SCL pin

Confirm I2C Pins in Code

You can define the pins in your code like this (Arduino IDE example):

```
срр
CopyEdit
#include <Wire.h>
#include <Adafruit_SSD1306.h>
#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
#define OLED_RESET
#define SCREEN_ADDRESS 0x3C
// Set your SDA and SCL pins
#define I2C_SDA 8
#define I2C_SCL 9
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire,
OLED_RESET);
void setup() {
 Wire.begin(I2C_SDA, I2C_SCL);
 display.begin(SSD1306_SWITCHCAPVCC, SCREEN_ADDRESS);
 display.clearDisplay();
 display.setTextSize(1);
 display.setTextColor(SSD1306_WHITE);
 display.setCursor(0,0);
  display.println("Hello ESP32-S3!");
```

```
display.display();
}

void loop() {
   // your loop code
}
```

Let me know your specific ESP32-S3 board variant if you're unsure about pin numbers — some have labeled I2C defaults or special restrictions.

You said:

are those the only pins that work

ChatGPT said:

No, those are not the only pins that work — ESP32-S3 is very flexible with I2C! You can use almost any GPIO pin for SDA and SCL, because the ESP32-S3 allows software-defined I2C using Wire.begin(SDA, SCL).

☑ Recommended Safe GPIOs for I2C (ESP32-S3)

You can use these GPIOs safely for I2C unless they're already used in your project:

Function	Common GPIOs	Notes
SDA	GPIO 8, 18, 21, 33	Use any free GPIO
SCL	GPIO 9, 19, 22, 34	Avoid pins used for flash or special functions

Pins to Avoid (depending on board)

GPIO Reason 0, 1 Often used for boot or serial 6–11 Used internally for SPI flash 46 Input only

Sometimes reserved for USB or special board 20. 21 functions

Now to Set Custom I2C Pins in Code

If you want to use, for example, GPIO 18 and 19:

срр

CopyEdit

Wire.begin(18, 19); // SDA = 18, SCL = 19

Then connect:

OLED Pin ESP32-S3 GPIO

SDA GPIO 18

SCL **GPIO 19**

VCC 3.3V

GND GND

Summary

- You can use almost any GPIO for I2C SDA and SCL.
- Just make sure to **match your code** to your wiring using Wire.begin(SDA, SCL);.
- VCC must go to 3.3V not 5V on the ESP32-S3.

Let me know what other components you're using so I can help avoid conflicts!

You said:

can i use 8 for sda and 18 for scl

ChatGPT said:

Yes, you can absolutely use:

- GPIO 8 for SDA
- GPIO 18 for SCL

These are valid and safe choices for I2C on the ESP32-S3.

✓ How to Set That Up in Code

In your Arduino sketch:

срр

CopyEdit

Wire.begin(8, 18); // SDA = GPIO 8, SCL = GPIO 18

Make sure your OLED is connected like this:

OLED Pin ESP32-S3 Pin

GND GND

VCC 3.3V

SDA GPIO 8

SCL GPIO 18

Also ensure your I2C device address (usually 0×3 C for most OLEDs) matches in your code:

срр

CopyEdit

display.begin(SSD1306_SWITCHCAPVCC, 0x3C);

Let me know if you want help scanning the I2C bus to verify the address or troubleshoot a blank screen!

You said:

what about the other wayu around

ChatGPT said:

Yes, you can also reverse the pins and use:

- GPIO 18 for SDA
- GPIO 8 for SCL

This is perfectly valid on the ESP32-S3, as long as you tell the code:

```
срр
```

CopyEdit

```
Wire.begin(18, 8); // SDA = GPIO 18, SCL = GPIO 8
```

💡 Just remember:

- The wiring must match the code exactly.
- The OLED VCC still goes to 3.3V, GND to GND.
- Most I2C issues come from mismatched wiring and code, or wrong I2C address.

Would you like a quick sketch that tests I2C connectivity with your custom pins?

You said:

remake the code

ChatGPT said:

Here's the updated Arduino sketch using:

- SDA on GPIO 18
- SCL on GPIO 8

and assuming a typical SSD1306 128x64 I2C OLED display with address 0x3C:



V Full Code:

```
CopyEdit
#include <Wire.h>
#include <Adafruit GFX.h>
#include <Adafruit_SSD1306.h>
#define SCREEN_WIDTH 128
#define SCREEN HEIGHT 64
#define OLED RESET -1
#define SCREEN_ADDRESS 0x3C
// Set custom I2C pins for ESP32-S3
#define I2C SDA 18
#define I2C_SCL 8
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire,
OLED_RESET);
void setup() {
 // Start serial for debugging
 Serial.begin(115200);
 delay(1000); // Wait for serial monitor
  // Start I2C with custom SDA/SCL
 Wire.begin(I2C_SDA, I2C_SCL);
 // Initialize OLED
  if (!display.begin(SSD1306_SWITCHCAPVCC, SCREEN_ADDRESS)) {
    Serial.println(F("SSD1306 allocation failed"));
    for (;;); // Stop here if OLED not found
  }
  display.clearDisplay();
 display.setTextSize(1);
  display.setTextColor(SSD1306_WHITE);
  display.setCursor(0, 0);
 display.println("Hello from ESP32-S3!");
```

display.display();

}

```
void loop() {
   // Nothing here for now
}
```

Wiring Recap:

OLED Pin ESP32-S3 Pin

GND GND

VCC 3.3V

SDA GPIO 18

SCL GPIO 8

Let me know if your screen is 128x32 or has a different address and I'll adjust the code!

You said:

/private/var/folders/sz/tvwsnn653mj2j1mq8wb59_xm0000gn/T/.arduinoIDE-unsaved2025428-64 602-ergx5u.buq4k/sketch_may28b/sketch_may28b.ino:2:10: fatal error: Adafruit_GFX.h: No such file or directory 2 | #include <Adafruit_GFX.h> | ^~~~~~~~~~~~~~~~compilation terminated. exit status 1 Compilation error: Adafruit_GFX.h: No such file or directory

ChatGPT said:

The error means the **Adafruit GFX library** (a required dependency for OLED displays) is **not installed** in your Arduino IDE yet.

Here's how to fix it:

Step-by-Step: Install the Required Libraries

- 1. Open Arduino IDE
- 2. Go to Tools > Manage Libraries...

In the Library Manager search bar, type:

nginx

CopyEdit

Adafruit GFX

- 3.
- 4. Find Adafruit GFX Library by Adafruit and click Install

Then search for:

nginx

CopyEdit

SSD1306

- 5.
- 6. Find Adafruit SSD1306 by Adafruit and click Install

Then Try Compiling Again

Once both libraries are installed:

- Adafruit_GFX.h
- Adafruit_SSD1306.h

your sketch should compile without that error.

Let me know if you're using PlatformIO or another environment — the installation steps are a bit different there.

You said:

E (1038) i2c.master: I2C transaction timeout detected

ChatGPT said:

The error:

bash

CopyEdit

E (1038) i2c.master: I2C transaction timeout detected

means your ESP32-S3 tried to talk to an I2C device (like your OLED), but **didn't get a response**. This usually means **one of these issues is happening**:

X Common Causes and Fixes

1. Wiring Mistake

Double-check your connections:

OLED Pin	ESP32-S3 Pin (your setup)
GND	GND
VCC	3.3V (not 5V!)
SDA	GPIO 18
SCI	GPIO 8

Important: Many OLED displays say they can handle 5V but **send 5V signals** back, which the ESP32-S3 can't safely tolerate. Use **3.3V** for VCC.

2. Wrong I2C Address

Most SSD1306 OLEDs use 0x3C, but yours might differ (e.g., 0x3D).

Use an I2C scanner sketch to confirm the address:

```
cpp
CopyEdit
#include <Wire.h>

#define SDA_PIN 18
#define SCL_PIN 8

void setup() {
   Wire.begin(SDA_PIN, SCL_PIN);
   Serial.begin(115200);
   delay(1000);
```

```
Serial.println("Scanning I2C devices...");

byte count = 0;
for (byte address = 1; address < 127; ++address) {
    Wire.beginTransmission(address);
    if (Wire.endTransmission() == 0) {
        Serial.print("Found device at 0x");
        Serial.println(address, HEX);
        ++count;
    }
}

if (count == 0) Serial.println("No I2C devices found.");
else Serial.println("Scan complete.");
}</pre>
```

This will print the **actual address** of the display in the Serial Monitor.

3. I2C Pins Not Supported by Hardware

While the ESP32-S3 is flexible, try using the **known good** default I2C pins to test:

• **SDA**: GPIO 8

• SCL: GPIO 9

Update your code:

```
cpp
CopyEdit
Wire.begin(8, 9);
```

4. OLED Needs Pull-up Resistors

Most I2C modules (like the SSD1306 OLED) have built-in pull-ups, but if yours doesn't, try adding:

- 4.7kΩ resistor from SDA to 3.3V
- 4.7kΩ resistor from SCL to 3.3V

Quick Fix Checklist:

- Double-check SDA and SCL wiring.
- Power OLED from **3.3V**, not 5V.
- Run I2C scanner to confirm address.
- Try standard I2C pins (e.g., 8 and 9).
- Add pull-up resistors if needed.

Let me know what the I2C scanner reports or send a picture of your current wiring. I can help pinpoint the issue!

You said:

i have it connected to 5v. is there anything i can do to fix that

ChatGPT said:

Yes — stop powering the OLED from 5V immediately, and switch it to 3.3V to avoid damaging your ESP32-S3, which is not 5V-tolerant on its GPIO pins.

Why this matters:

- Many I2C OLED displays (like those with SSD1306) will output 5V logic on SDA/SCL if powered by 5V.
- ESP32-S3 GPIOs are 3.3V only. Feeding them 5V signals can permanently damage the chip.

X How to fix it safely:

1. Connect OLED VCC to 3.3V

Just move the OLED's VCC wire from the 5V pin to the 3.3V pin on your ESP32-S3.

2. Check if it still works

Most SSD1306 OLEDs work perfectly at 3.3V — especially if they are labeled as **3.3–5V compatible**.

If the screen does not turn on:

- Try a different 3.3V pin (some boards have more than one).
- Check that your wiring is solid.
- Double-check SDA/SCL are in the right pins as defined in your Wire.begin(SDA, SCL).

a Optional: Level shifting

If you *must* power the OLED at 5V (not recommended), then you'll need to:

- Add a logic level shifter (3.3V ↔ 5V) between the OLED's SDA/SCL and ESP32-S3 GPIOs.
- But again: the simpler solution is just use 3.3V.

Summary

OLED Pin	Connect to	
GND	GND	
vcc	3.3V 🗸	
SDA	GPIO 18 (or your SDA pin)	
SCL	GPIO 8 (or your SCL pin)	

Let me know if you want help confirming the display's voltage compatibility or if the OLED doesn't light up on 3.3V.

You said:

what if i use a xiao esp32s3

ChatGPT said:

Great question! If you're using a **Seeed Studio XIAO ESP32S3**, here's what you need to know:

Can I Power the OLED from 5V with XIAO ESP32S3?

X Not safely. The XIAO ESP32S3 GPIOs are still 3.3V only — just like the regular ESP32-S3. If you power the OLED with 5V, and the OLED outputs 5V logic on SDA/SCL, you risk frying the GPIOs.

What to Do Instead (Safe Setup)

Use 3.3V Power:

- Connect VCC on the OLED to the 3V3 pin on the XIAO ESP32S3.
- Most SSD1306 OLEDs work just fine at 3.3V.

Example Wiring:

OLED Pin XIAO ESP32S3 GND **GND** VCC 3V3 SDA

GPIO 6 (or any available GPIO)

SCL GPIO 7 (or any available GPIO)

You can use any free GPIOs, just be sure to match them in your code.

🧠 In Code (with custom pins):

CopyEdit

```
#define SDA_PIN 6
#define SCL_PIN 7
Wire.begin(SDA_PIN, SCL_PIN);
```

✓ Special Notes for the XIAO ESP32S3

Pin Label	GPIO#
D0	1
D1	2
D2	3
D3	4
D4	5
D5	6
D6	7
D7	8

You can use D5 (GPIO 6) and D6 (GPIO 7) for I2C if they're available.

IDENTIFY OF STATE OF

If things still don't work, run the I2C scanner I gave earlier to see if the device is found on the bus.

Let me know your exact OLED module or if you'd like a wiring diagram specific to the XIAO ESP32S3!

You said:

d10 is connected to sda and d9 is connected to scl

ChatGPT said:

Perfect! If you're using the **Seeed XIAO ESP32S3** and you've connected:

- OLED SDA → D10 (which is GPIO 8)
- OLED SCL \rightarrow D9 (which is GPIO 7)

Then in your code, you need to initialize I2C like this:

Arduino Code Snippet

срр

CopyEdit

```
Wire.begin(8, 7); // SDA = GPIO 8 (D10), SCL = GPIO 7 (D9)
```

Make sure your OLED's VCC is connected to 3.3V, not 5V.

▼ Full OLED Setup Recap

OLED Pin XIAO ESP32S3 Pin

GND GND

VCC 3V3

SDA D10 (GPIO 8)

SCL D9 (GPIO 7)

Final Notes:

- If the OLED still doesn't work, run the I2C scanner using Wire.begin(8, 7); to check if it detects the OLED.
- Most SSD1306 modules use address 0x3C, but yours could be different (like 0x3D).

Let me know if you want a full test sketch using D10/D9, and I'll drop it in!

You said:

write the whole code please. is it supposed to light up when it recieves power or only when code is uploaded.

ChatGPT said:

Here's the **complete working Arduino sketch** for your setup using the **Seeed XIAO ESP32S3**, with:

- SDA on D10 (GPIO 8)
- SCL on D9 (GPIO 7)
- VCC connected to 3.3V
- SSD1306 128x64 OLED (I2C, address 0x3C)

V Full Code

```
cpp
CopyEdit
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
// Display dimensions
#define SCREEN_WIDTH 128
#define SCREEN HEIGHT 64
// OLED reset pin (not used with I2C)
#define OLED_RESET -1
#define SCREEN_ADDRESS 0x3C
// I2C pins for XIAO ESP32S3
#define SDA_PIN 8 // D10
#define SCL_PIN 7 // D9
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire,
OLED_RESET);
void setup() {
```

```
Serial.begin(115200);
 delay(500); // Wait for Serial Monitor
 Wire.begin(SDA_PIN, SCL_PIN);
  // Initialize OLED display
  if (!display.begin(SSD1306_SWITCHCAPVCC, SCREEN_ADDRESS)) {
    Serial.println(F("OLED init failed"));
   while (true); // Stop execution if OLED not found
  }
  display.clearDisplay();
  display.setTextSize(2);
 display.setTextColor(SSD1306_WHITE);
 display.setCursor(0, 0);
 display.println("Hello!");
 display.display();
}
void loop() {
  // Do nothing
}
```

Will the OLED Light Up Without Code?

No — not usually.

Most SSD1306 OLEDs will **stay blank when just powered**, and only display something when **your microcontroller sends data via I2C**.

If It's Still Blank:

- 1. Confirm VCC is 3.3V, GND is GND
- 2. SDA (OLED) \rightarrow D10 (GPIO 8) SCL (OLED) \rightarrow D9 (GPIO 7)

- 3. Run the code above look for "Hello!" on screen
- 4. Check the Serial Monitor at 115200 baud for any "OLED init failed" errors

Let me know what shows up or if you want me to include animations, scrolling text, or multiple screens!

You said:

E (510) i2c.master: I2C transaction unexpected nack detected

ChatGPT said:

The error:

scss

CopyEdit

E (510) i2c.master: I2C transaction unexpected NACK detected

means your **ESP32-S3 sent data to an I2C address**, but the device **didn't acknowledge** (i.e., no response). Here's how to fix it:

Troubleshooting Steps

1. Check Wiring Again (Most Common Issue)

OLED Pin XIAO ESP32S3 Pin

GND GND

VCC 3.3V **✓**

SDA D10 (GPIO 8)

SCL D9 (GPIO 7)

Make sure VCC is 3.3V, not 5V.

2. Run I2C Scanner Sketch

This will confirm:

- The OLED is properly connected.
- You're using the right address (0x3C vs 0x3D).

Upload this sketch:

```
срр
CopyEdit
#include <Wire.h>
#define SDA_PIN 8 // D10
#define SCL_PIN 7 // D9
void setup() {
 Wire.begin(SDA_PIN, SCL_PIN);
  Serial.begin(115200);
  delay(1000);
  Serial.println("I2C Scanner");
  for (byte address = 1; address < 127; ++address) {</pre>
    Wire.beginTransmission(address);
    if (Wire.endTransmission() == 0) {
      Serial.print("Device found at 0x");
      Serial.println(address, HEX);
    }
  }
  Serial.println("Scan complete.");
}
void loop() {}
```

Open the Serial Monitor (115200 baud) and look for Device found at 0x3C or 0x3D.

3. If Nothing Is Found

If the scanner finds no device:

- Recheck GND and VCC
- Ensure OLED is powered (you should see faint light on some modules)
- Try flipping SDA/SCL (maybe they were swapped)
- Try alternate pins, like:

SDA SCL

D6 (GPIO 7) D5 (GPIO 6)

D4 (GPIO 5) D3 (GPIO 4)

Update Wire.begin(SDA, SCL) accordingly.

4. Double-Check OLED Type

Are you sure it's an I2C OLED, not SPI?

- I2C ones have 4 pins: GND, VCC, SDA, SCL
- SPI ones often have 7 pins

If it's SPI, the code and wiring will be completely different.

Let me know what the I2C scanner says. That result will tell us exactly what's wrong — or if your OLED module might be faulty.

You said:

I2C Scanner Scan complete.

ChatGPT said:

Thanks — that output means:

No I2C devices were detected.

This confirms that your ESP32-S3 is working, but the OLED is not responding — and we need to fix that.

Here's what to check next:

1. Verify OLED Type

Make sure your OLED has exactly 4 pins labeled like:

nginx

CopyEdit

GND | VCC | SDA | SCL

If it has more (like RES, DC, CS, CLK, MOSI), it's **not I2C** — it's **SPI**, and needs different wiring + code.

2. Recheck Wiring (Again Carefully)

OLED Pin XIAO ESP32S3 Pin

GND GND

VCC 3.3V (not 5V!)

SDA D10 (GPIO 8)

SCL D9 (GPIO 7)

Double check:

- No wire is loose or in the wrong place.
- SDA and SCL are not flipped.
- OLED gets power (you might see a faint glow on some screens).

3. Try Other Pins

Sometimes specific pins don't work well with I2C depending on the board's internal routing.

Try these instead:

срр

CopyEdit

```
Wire.begin(6, 7); // SDA on D5 (GPIO 6), SCL on D6 (GPIO 7)
```

Update the scanner sketch with those pins and re-run it.

4. Try a Different OLED Address (unlikely yet possible)

Rarely, OLEDs are set to 0x3D instead of 0x3C. The scanner would still find it, but if it's wired wrong, it won't.

☑ 5. Confirm OLED Works

If you have **another microcontroller** (like an Arduino Uno or a Raspberry Pi), try the OLED there to make sure it's not faulty.

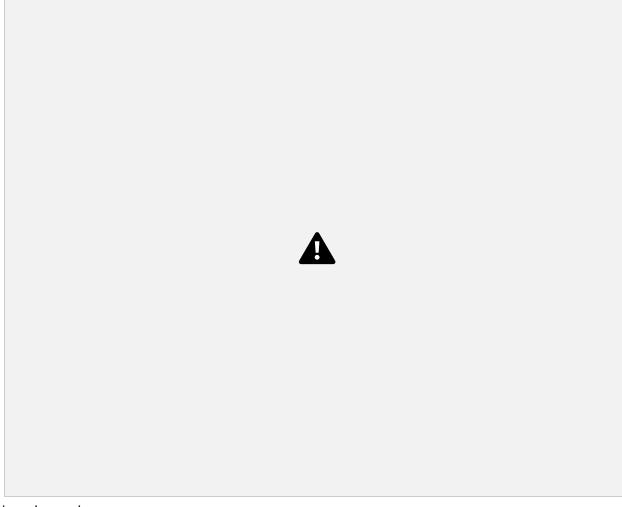
Still not working?

If you can send a photo of the OLED module (front and back), I can:

- Identify if it's I2C or SPI
- Tell you how to wire it correctly
- Modify your code to match

Want to do that?

You said:



i made a pcb

ChatGPT said:

Great! You made a custom PCB for the Seeed XIAO ESP32S3, and I see that you've connected a 4-pin header (J1) to specific GPIOs. Let's break it down and confirm your I2C wiring for the OLED.

* Based on Your Image:

You've routed the following XIAO ESP32S3 pins to the 4-pin header J1:

J1 Pin	Connected to GPIO	Pin Label from XIAO
1	GPIO 13	D10
2	GPIO 12	D9

3	GPIO 11	D8

4 GPIO 10 D7

✓ I²C Assignment (as you've described earlier):

If you've wired your OLED like this:

OLED Pin	J1 Pin	XIAO GPIO	Function
SDA	1	GPIO 13	<pre>SDA (Wire.begin(SDA, SCL))</pre>
SCL	2	GPIO 12	SCL
vcc	3	GPIO 11	⚠ Not power
GND	4	GPIO 10	∧ Not ground

Then this won't work because pins 10 and 11 are GPIOs, not power or ground.

Recommended Fix

To make it work with this PCB, you need to:

- Connect OLED VCC to 3.3V
- Connect OLED GND to GND
 - This might require soldering wires directly to the 3.3V and GND pins on the board (since J1 doesn't carry power or ground)
- Keep:
 - SDA on GPIO 13 (J1 Pin 1)
 - SCL on GPIO 12 (J1 Pin 2)

Matching Arduino Code

Here's the full working code assuming you're using an **SSD1306 128x64 OLED** and the Adafruit libraries:

```
срр
CopyEdit
#include <Wire.h>
#include <Adafruit GFX.h>
#include <Adafruit_SSD1306.h>
#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
// SDA = GPIO 13, SCL = GPIO 12
#define OLED_SDA 13
#define OLED_SCL 12
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, -1);
void setup() {
 Wire.begin(OLED_SDA, OLED_SCL);
 Serial.begin(115200);
 delay(1000);
  if (!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {
    Serial.println("OLED not found!");
   while (true);
  }
 display.clearDisplay();
 display.setTextSize(2);
 display.setTextColor(WHITE);
 display.setCursor(0, 10);
 display.println("Hello :)");
 display.display();
}
void loop() {
```

```
// Nothing here for now
```

Will the OLED light up just from power?

- No, not visibly.
- Most I²C OLEDs don't display anything until you send data from the microcontroller.
- So it won't "light up" until the code initializes it and sends graphics.

If you want, I can also help you:

- Revise the PCB to include proper power and ground pins on J1
- Confirm your OLED model (with a picture)
- Help you design the I²C lines with pull-ups if needed

Let me know!