

ADD WI-FI TO YOUR AIS DEVICE

DIY NMEA 0183 to Wi-Fi Bridge Setup Guide

There is no need to pay hundreds of dollars for commercial NMEA to Wi-Fi multiplexers. If you already have an old or new AIS device (Matsutec, Raymarine, Em-Trak, etc.) on your boat, we will set up a simple \$10 system that takes the data from its cable (NMEA OUT) and broadcasts it wirelessly over Wi-Fi.

This system allows OrSoft Mariner PRO to catch marine signals with zero latency and run the collision avoidance system (CPA/TCPA) completely offline, independent of the outside world.

STEP 1: Shopping List (Hardware) We only need 3 small and inexpensive electronic parts:

1. **ESP8266 Module (NodeMCU):** The brain and Wi-Fi antenna of our system. Costs around 4-5 USD.
2. **MAX485 Module (Signal Translator):** Translates the marine standard RS422 voltage into a low voltage (TTL) that our ESP brain can understand. It is the size of a fingernail and costs 1-2 USD.
3. **12V/24V to 5V DC-DC Buck Converter:** The power module required to safely draw 5V power from the boat battery.

STEP 2: Wiring The diagram for taking the marine signal and transferring it to the microchip is quite simple. Connect the pins as follows:

- **From Your AIS Device:** Connect the NMEA OUT+ (TX+) wire from the AIS to the **A** pin of the MAX485, and the NMEA OUT- (TX-) wire to the **B** pin of the MAX485.
- **From MAX485 Module to ESP8266:**
 - MAX485 VCC → ESP8266 3V3 (3.3 Volts Power)
 - MAX485 GND → ESP8266 GND (Ground)
 - MAX485 RO → ESP8266 RX (Reading Pin)
 - **Important:** Bridge the **RE** and **DE** pins on the MAX485 together and connect them directly to **GND**. (This keeps the module permanently in "Listening" mode).

STEP 3: Uploading Software to the Chip Connect the ESP8266 chip to your computer using a USB cable. Download the Arduino IDE program (arduino.cc/en/software). Select "NodeMCU 1.0 (ESP-12E)" as the board in the program settings. Then copy the entire code below, paste it onto the screen, and press the **Upload** button.

C++

```
#include <ESP8266WiFi.h>
#include <WiFiUdp.h>
```

```
const char *ssid = "OrSoft_AIS_Bridge";
const char *password = "12345678";
const int udpPort = 10110;
```

```
WiFiUDP udp;
```

```
void setup() {
```

```

// AIS devices generally use a baud rate of 38400.
Serial.begin(38400);

// Creating our own Wi-Fi network (Works even if there's no router on the
boat)
WiFi.softAP(ssid, password);

// Opening the UDP Broadcast port
udp.begin(udpPort);
}

void loop() {
  if (Serial.available()) {
    String data = Serial.readStringUntil('\n');

    // Broadcast the data over the air (Send to all connected devices)
    udp.beginPacket("192.168.4.255", udpPort);
    udp.print(data + "\n");
    udp.endPacket();
  }
}

```

As soon as you see the "Done uploading" text on the screen, you can unplug your chip from the computer. It is now a marine device!

STEP 4: Connecting to OrSoft Mariner PRO Once you connect the assembled device to your boat battery and power it up, everything is ready.

1. Go to your mobile phone's Wi-Fi settings and connect to the **OrSoft_AIS_Bridge** network (Password: 12345678).
2. Open the OrSoft Mariner PRO app and enter the **Settings** menu.
3. Select "**Internal Data (Wi-Fi/NMEA)**" as the Data Source.
4. Type **192.168.4.1** in the IP Address section (This is the fixed address of the ESP chip).
5. Type **10110** in the UDP Port section and press Save.

Congratulations, Captain! Your old, screenless AIS device now gives life to your phone with the most modern technology. Your collision avoidance system is at your command! Fair winds.