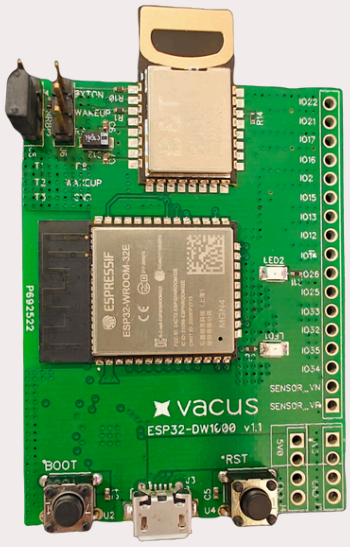
VACUS TECH

# Explore the Future of Location Precision with the ESP-DW1000 Evaluation Board

## Your Journey towards Innovation Begins with the ESP32-DW1000



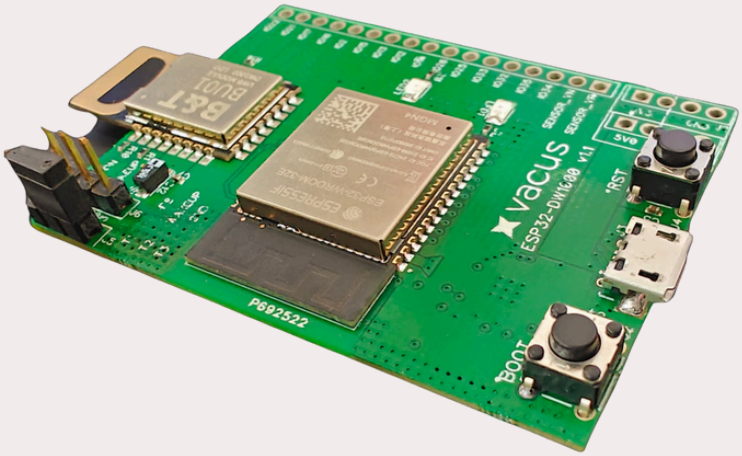




Fig: Image of ESP32-DW1000

# Table of Content

## 1 Demo Video

- Click on the link below to open the video.
- Copy and Paste the URL into the browser.

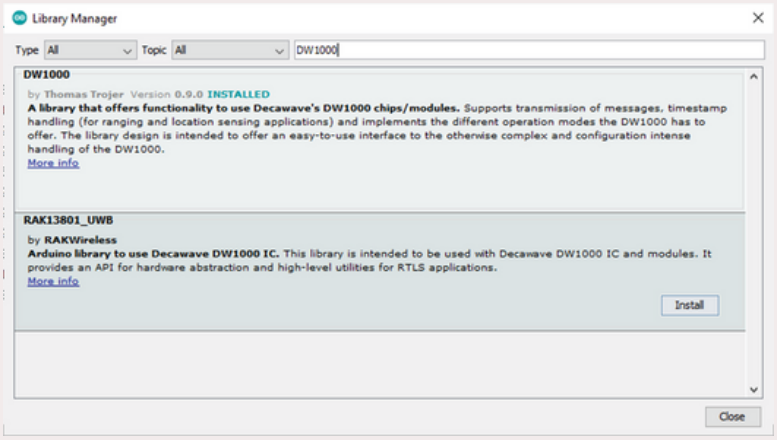
https://drive.google.com/file/d/1iL8BeEW0ehmeyeVX73UecmaHv_SIwtUk/view

## 2. Using the ESP32 DW1000 UWB Board with Arduino IDE

In this guide, we'll learn how to use the ESP32 DW1000 UWB (Ultra Wideband) board with the Arduino IDE to measure the distance between two boards. For this project, you'll need a pair of boards. We'll follow a series of steps to set up the module.

### 2.1 Installing the DW1000 Library

First, you'll need to install the Arduino-DW1000 library from Thotro. This library provides the necessary functionality to work with Decawave's DW1000 chips and modules in the Arduino environment.

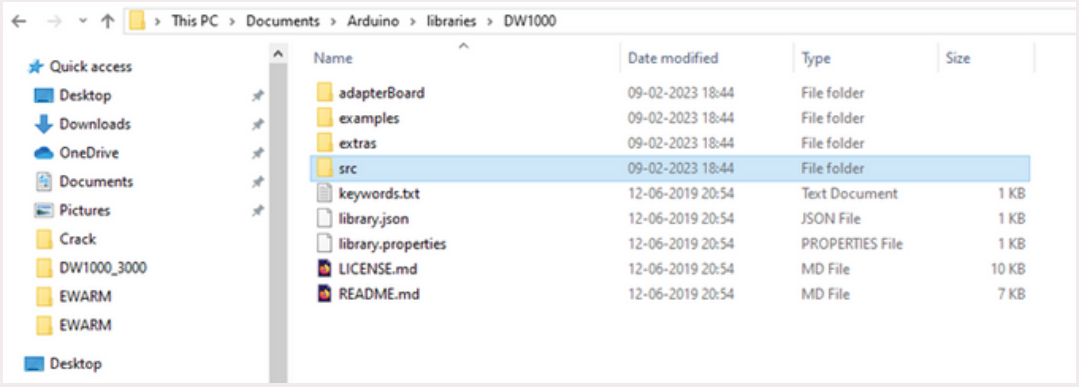



You can also install this library using the Library Manager. Simply search for "DW1000" and click "Install" to add the library to your Arduino IDE.

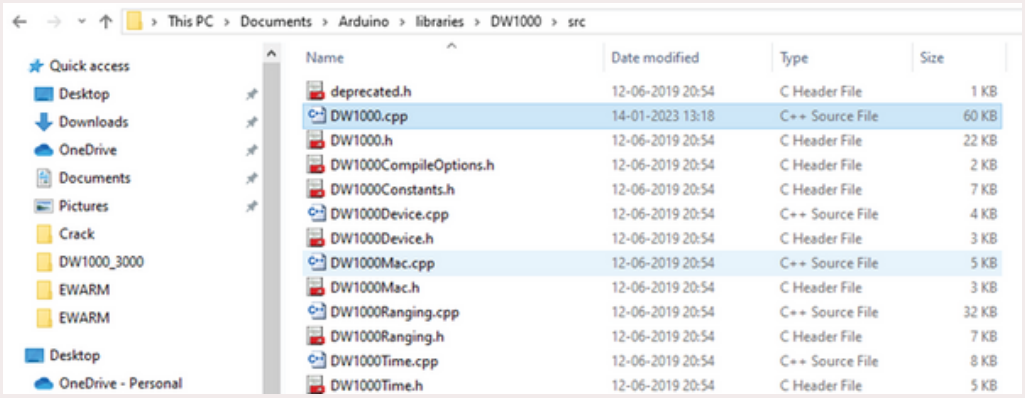
### 2.2 Modifying the Library

The DW1000 UWB library doesn't compile directly for ESP32 boards, so we need to make some modifications.

First, navigate to the Arduino library folder and locate the DW1000 folder. Inside that folder, open the "src" folder to access the library's source files.




Open the "src" folder and find the DW1000.cpp file. Use a text editor, such as Notepad++, to open this file.




Next, locate the following lines (Line 172) and comment out all three lines.

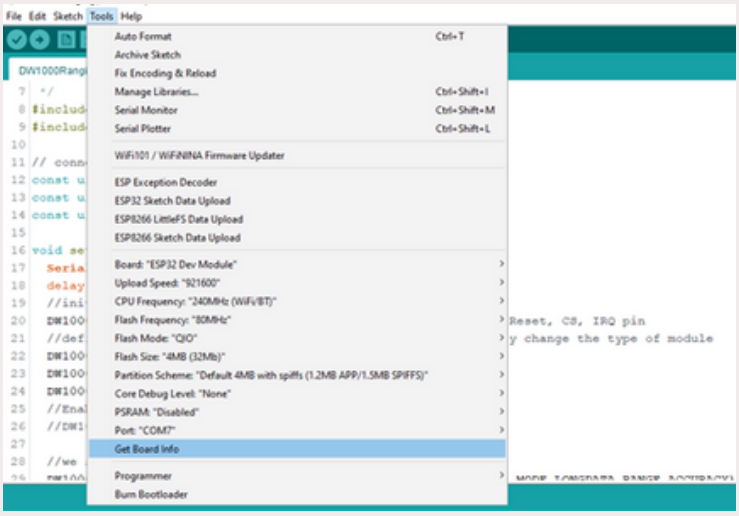
```
172 /* #ifndef ESP8266
173     SPI.usingInterrupt(digitalPinToInterrupt(irq)); // not every board support this,
174 #endif */
```

```
163 }
164
165 void DW1000Class::begin(uint8_t irq, uint8_t rst) {
166     // generous initial init/wake-up-idle delay
167     delay(5);
168     // Configure the IRQ pin as INPUT. Required for correct interrupt setting for ESP8266
169         pinMode(irq, INPUT);
170     // start SPI
171     SPI.begin();
172 /* #ifndef ESP8266
173     SPI.usingInterrupt(digitalPinToInterrupt(irq)); // not every board support this, e.g. ESP8266
174 #endif */
175     // pin and basic member setup
176     _rst        = rst;
177     _irq        = irq;
178     _deviceMode = IDLE_MODE;
179     // attach interrupt
180     //attachInterrupt(_irq, DW1000Class::handleInterrupt, CHANGE); // todo interrupt for ESP8266
181     // TODO throw error if pin is not a interrupt pin
182     attachInterrupt(digitalPinToInterrupt(_irq), DW1000Class::handleInterrupt, RISING); // todo i
183 }
```

Once these lines are commented out, the library code will compile successfully.

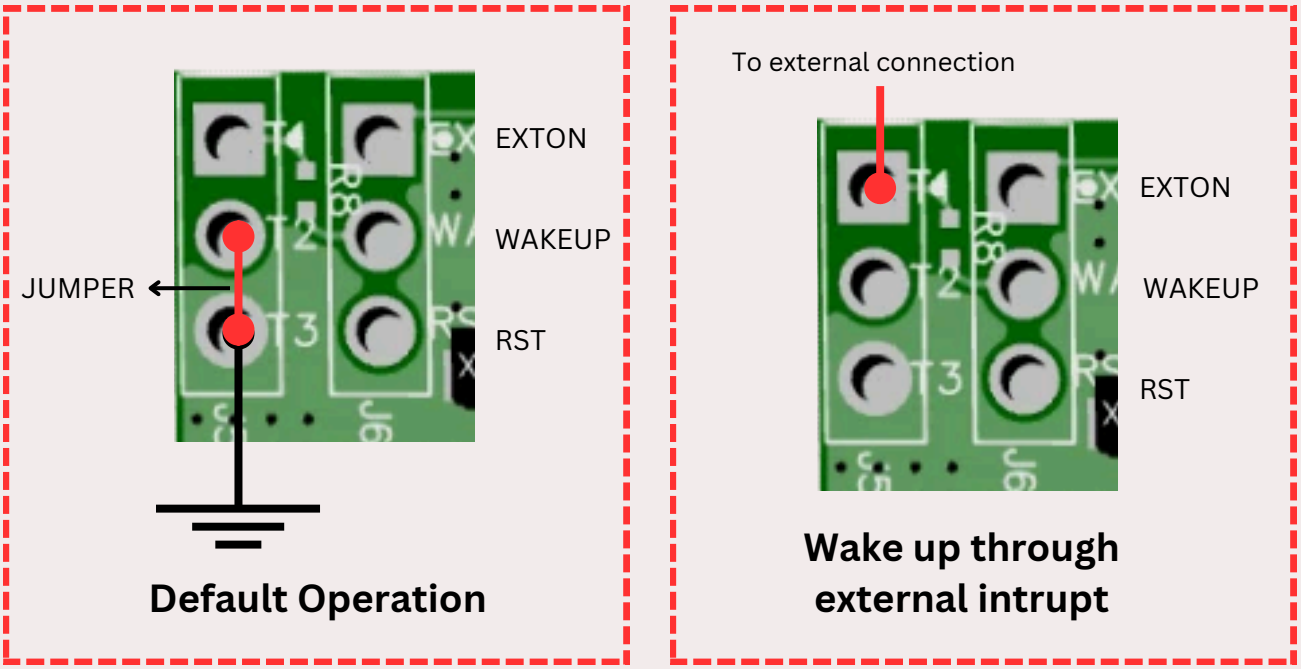
## 2.3 Board Selection

Connect the pair of ESP32 Wrover boards to two different USB ports on your computer using micro-USB cables. In the Arduino IDE, select the development board: choose "ESP32 Dev Module" if you are using the ESP32 UWB board with the ESP32 WROOM chip. If you have the ESP32 UWB board with the ESP32 WROVER chip, select "ESP32 WROVER Module."




Also, make sure to select the correct COM port, which you can find in the Device Manager. Your ESP32 Ultra Wideband board is now set up for serial communication.

## 3. Hardware




### Pin connection

| PIN_RST | IO 27 |
|---|---|
| IRQ | IO 4 |
| SS | IO 5 |
| LED 1 | IO 17 |
| LED 2 | IO 26 |