BIP DIG-SENSING

ESP32 Get started - reminder

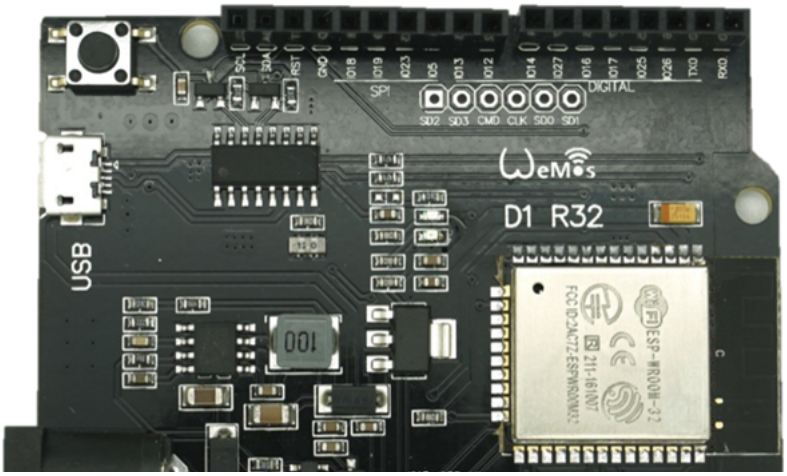
Introduction:

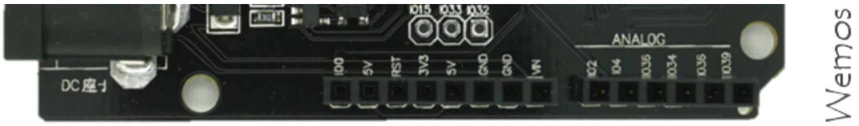
To use the ESP32 board you need to have the Arduino IDE program installed, and the ES32 board configured. In this project we learn how to use and configure it.

What is Arduino?

Arduino is an open source hardware and software platform designed to make electronics more accessible to general public. Arduino programming is done using a C++-based programming language, and a wide variety of sensors, actuators, and other electronic components can be accessed through open source software libraries.

ESP32 BOARD (our board):

What is ESP32? :	ESP32 is a low-cost, low-power system in a chip (SoC), Created by Espressif Systems, this chip has integrated series with Wi-Fi & dual-mode Bluetooth capabilities.
ESP32 Board:	 USB WeMos D1 R32 DIGITAL SPI SD2 SD3 CMD CLK SD0 SD1 ESP-WROOM-32 D1 R32 (based on ESP32)

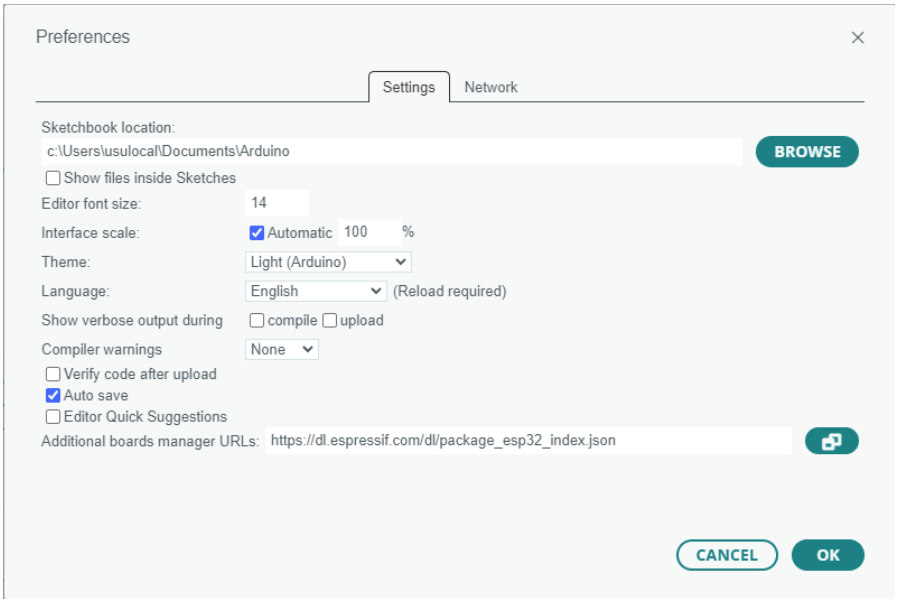
	
Links of interest:	**Download Arduino IDE 2.0.4: Download** **Download Required drivers: Download (Don't use at the moment)**

HOW TO CONFIGURITE ESP32 BOARD:

Step by Step:

Step 1

Open the Arduino IDE and go to the "File" menu, then "Preferences". In the "Additional Board Manager URLs" field, paste the following URL: https://dl.espressif.com/dl/package_esp32_index.json



Preferences

Sketchbook location:
c:\Users\usulocal\Documents\Arduino

□ Show files inside Sketches

Editor font size: 14

Interface scale: ☑ Automatic 100 %

Theme: Light (Arduino)

Language: English (Reload required)

Show verbose output during □ compile □ upload

Compiler warnings None

□ Verify code after upload

☑ Auto save

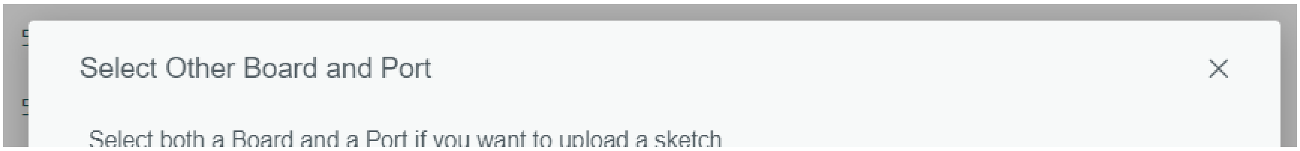
□ Editor Quick Suggestions

Additional boards manager URLs: https://dl.espressif.com/dl/package_esp32_index.json

Step 2

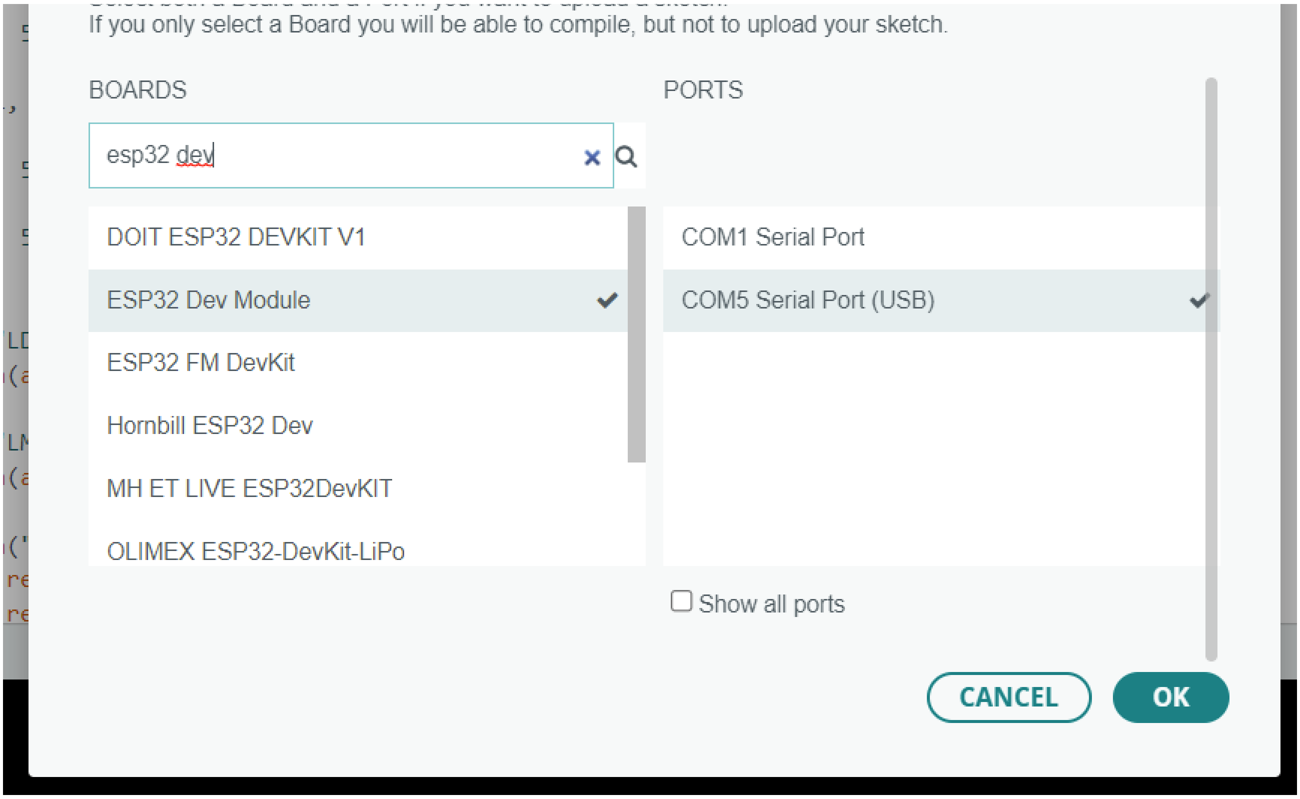
Now, select the port where we have the board plugged in.

Above we will get this drop-down, we have to choose the board and get the following pop-up window:



Select Other Board and Port

Select both a Board and a Port if you want to upload a sketch



If you only select a Board you will be able to compile, but not to upload your sketch.

BOARDS

esp32 dev

- DOIT ESP32 DEVKIT V1
- ESP32 Dev Module ✓
- ESP32 FM DevKit
- Hornbill ESP32 Dev
- MH ET LIVE ESP32DevKIT
- OLIMEX ESP32-DevKit-LiPo

PORTS

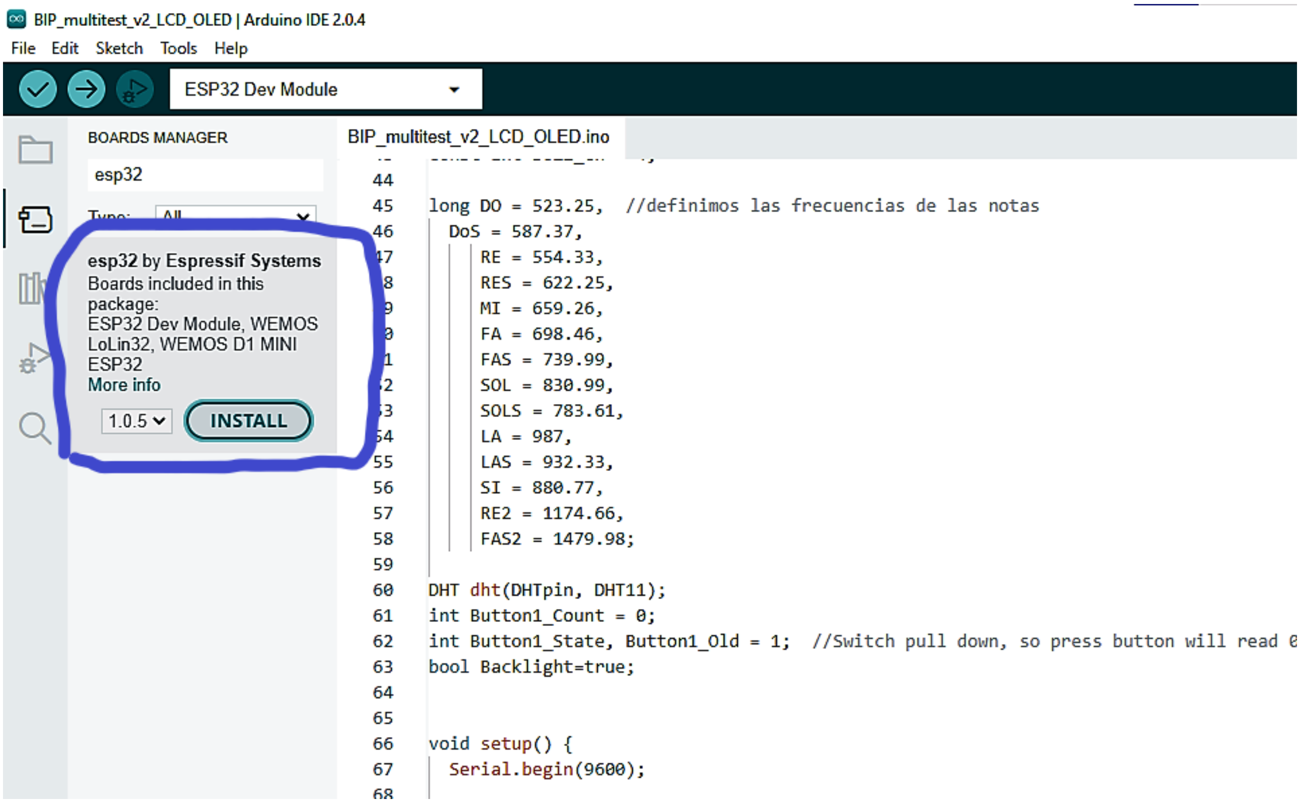
- COM1 Serial Port
- COM5 Serial Port (USB) ✓

□ Show all ports

Write in the search engine "ESP32 Dev" we select "ESP32 Dev Module". Click on OK

Step 3

Go to the "Tools" menu, then "Board: " Click on "Boards Manager". In the search bar, type "esp32" and select the "esp32 by Espressif Systems" package and Install.



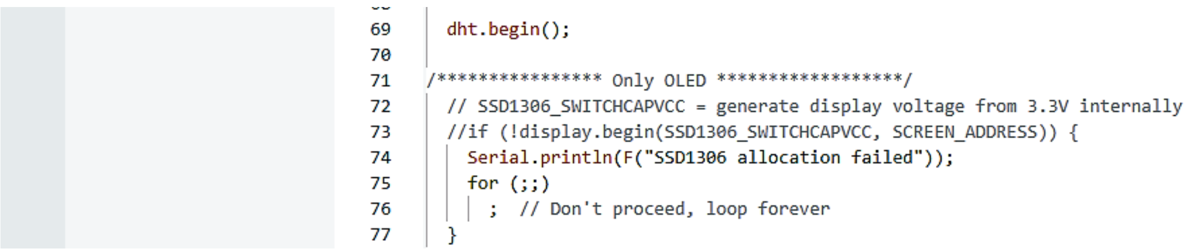
BIP_multitest_v2_LCD_OLED | Arduino IDE 2.0.4
File Edit Sketch Tools Help
ESP32 Dev Module
BOARDS MANAGER
BIP_multitest_v2_LCD_OLED.ino

esp32

esp32 by Espressif Systems
Boards included in this package:
ESP32 Dev Module, WEMOS LoLin32, WEMOS D1 MINI ESP32
More info

1.0.5 INSTALL

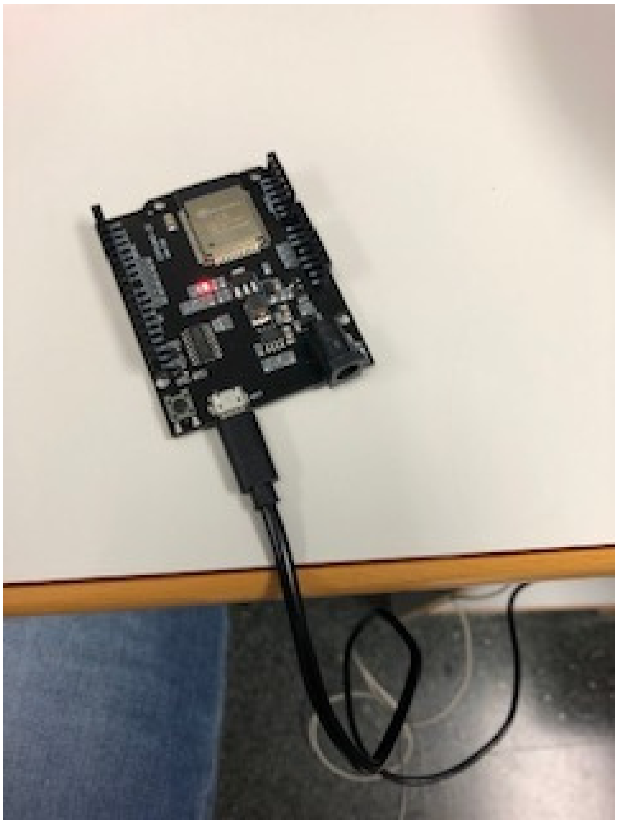
```
long DO = 523.25,  //definimos las frecuencias de las notas
  DoS = 587.37,
     RE = 554.33,
     RES = 622.25,
     MI = 659.26,
     FA = 698.46,
     FAS = 739.99,
     SOL = 830.99,
     SOLS = 783.61,
     LA = 987,
     LAS = 932.33,
     SI = 880.77,
     RE2 = 1174.66,
     FAS2 = 1479.98;

DHT dht(DHTpin, DHT11);
int Button1_Count = 0;
int Button1_State, Button1_Old = 1;  //Switch pull down, so press button will read 
bool Backlight=true;


void setup() {
  Serial.begin(9600);
```



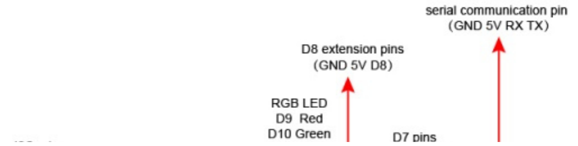
```
  dht.begin();

/*************** Only OLED *****************/
  // SSD1306_SWITCHCAPVCC = generate display voltage from 3.3V internally
  //if (!display.begin(SSD1306_SWITCHCAPVCC, SCREEN_ADDRESS)) {
    Serial.println(F("SSD1306 allocation failed"));
    for (;;)
      ;  // Don't proceed, loop forever
  }
```

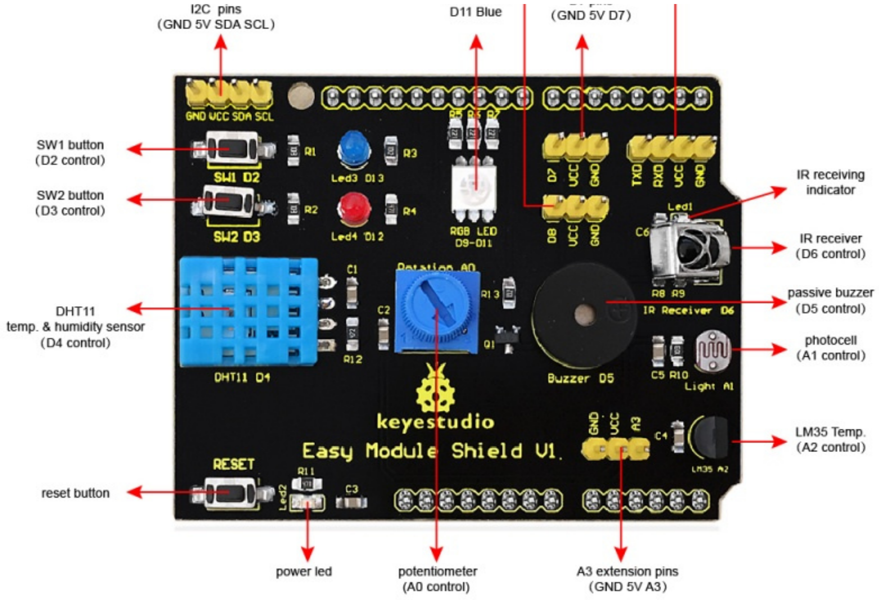
Step 4

Click on the "Install" button and wait for the installation to finish. Connect the ESP32 board to your computer via a USB cable. Go to Tools > Port and select the port that the ESP32 is connected to. Connect the board like this:



MULTISENSOR

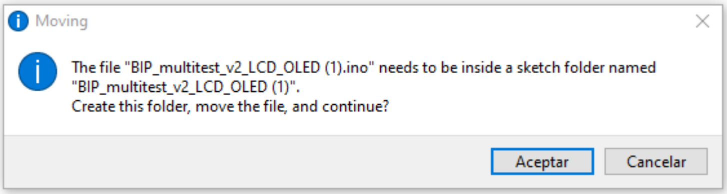
What is a Multisensor?	A multisensor shield for Arduino ESP32 is a device that combines several sensors on a single board, it can measure different variables, such as temperature, humidity, light, sound, movement, etc
Multisensor Board and pinned :	 serial communication pin (GND 5V RX TX) D8 extension pins (GND 5V D8) RGB LED D9 Red D10 Green D7 pins

	
Link of interest:	**Video installing necessary libraries: Video**

HOW TO ACTIVATE EACH INTEGRATED BLOCK

Step 1

First we download the pin multisensor checker code, available in this link Click on "accept"

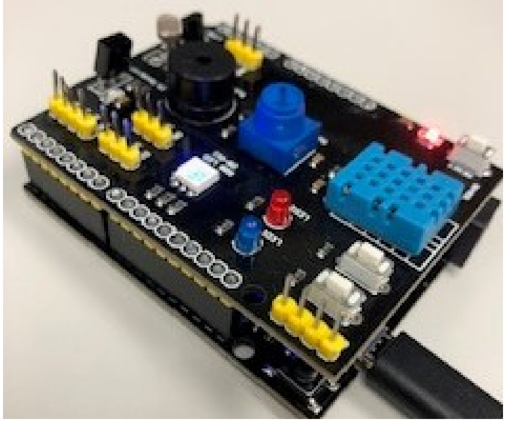


Moving

The file "BIP_multitest_v2_LCD_OLED (1).ino" needs to be inside a sketch folder named "BIP_multitest_v2_LCD_OLED (1)".
Create this folder, move the file, and continue?

Step 2

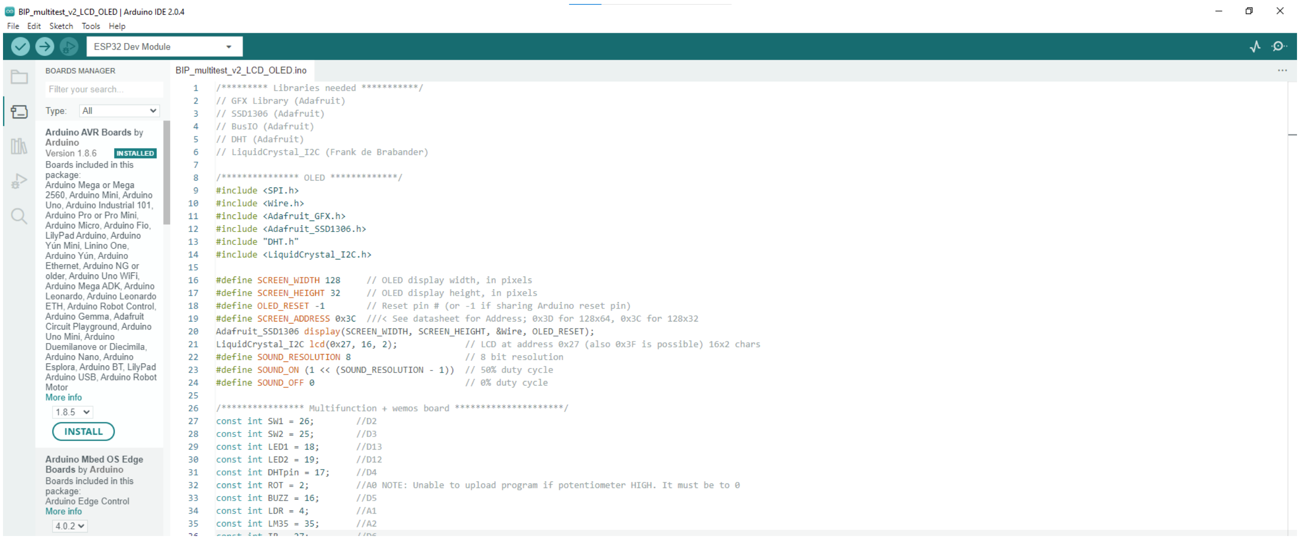
Plug the multisensor board to the ESP32 board and then we connect the USB cable to your PC.



It is possible that the Esp32 microcontroller was previously programmed with some funcions. It doesn't matter.

Step 3

We proceed now to open a pin multisensor checker code, available in this link, and select our board and parameters as we decribed above.



BIP_multitest_v2_LCD_OLED | Arduino IDE 2.0.4

ESP32 Dev Module

Type: All

Arduino AVR Boards by **Arduino**
Version 1.8.6
Boards included in this package:
Arduino Mega or Mega 2560, Arduino Mini, Arduino Uno, Arduino Industrial 101, Arduino Pro or Pro Mini, Arduino Micro, Arduino Fio, LilyPad Arduino, Arduino Yún Mini, Linino One, Arduino Yún, Arduino Ethernet, Arduino NG or older, Arduino Uno WiFi, Arduino Mega ADK, Arduino Leonardo, Arduino Leonardo ETH, Arduino Robot Control, Arduino Gemma, Adafruit Circuit Playground, Arduino Uno Mini, Arduino Duemilanove or Diecimila, Arduino Nano, Arduino Esplora, Arduino BT, LilyPad Arduino USB, Arduino Robot Motor
1.8.5

Arduino Mbed OS Edge Boards by **Arduino**
Boards included in this package:
Arduino Edge Control
4.0.2

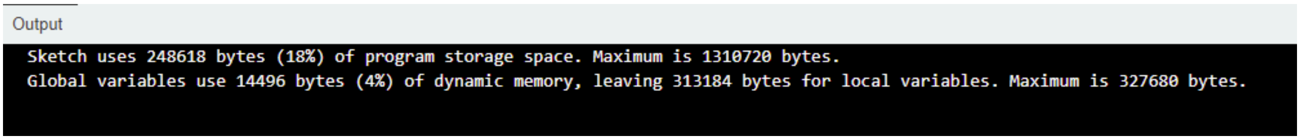
BIP_multitest_v2_LCD_OLED.ino

```
/********* Libraries needed ***********/
// GFX Library (Adafruit)
// SSD1306 (Adafruit)
// BusIO (Adafruit)
// DHT (Adafruit)
// LiquidCrystal_I2C (Frank de Brabander)

/*************** OLED *************/
#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include "DHT.h"
#include <LiquidCrystal_I2C.h>

#define SCREEN_WIDTH 128     // OLED display width, in pixels
#define SCREEN_HEIGHT 32     // OLED display height, in pixels
#define OLED_RESET -1        // Reset pin # (or -1 if sharing Arduino reset pin)
#define SCREEN_ADDRESS 0x3C  ///< See datasheet for Address; 0x3D for 128x64, 0x3C for 128x32
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);
LiquidCrystal_I2C lcd(0x27, 16, 2);            // LCD at address 0x27 (also 0x3F is possible) 16x2 chars
#define SOUND_RESOLUTION 8                     // 8 bit resolution
#define SOUND_ON (1 << (SOUND_RESOLUTION - 1))  // 50% duty cycle
#define SOUND_OFF 0                            // 0% duty cycle

/*************** Multifunction + wemos board ********************/
const int SW1 = 26;        //D2
const int SW2 = 25;        //D3
const int LED1 = 18;       //D13
const int LED2 = 19;       //D12
const int DHTpin = 17;     //D4
const int ROT = 2;         //A0 NOTE: Unable to upload program if potentiometer HIGH. It must be to 0
const int BUZZ = 16;       //D5
const int LDR = 4;         //A1
const int LM35 = 35;       //A2
```

See the following video to check the board and libraries. IMPORTANT: chose the 115200 "Upload Speed" in Tools menu. **Video**

Step 4

Proceed to compile the code (top left button) and verify no-errors. The following message should appears in your window.



Output

```
Sketch uses 248618 bytes (18%) of program storage space. Maximum is 1310720 bytes.
Global variables use 14496 bytes (4%) of dynamic memory, leaving 313184 bytes for local variables. Maximum is 327680 bytes.
```

The message shows you the required memory space and additional information.

Step 5

The "multisensor pin checher" includes several sensors. Analize them and try to understand the lines useful for each of these sensors. For example, we identify the piece of code useful to use a LCD/OLED attached to the SPI (not included in the board) or the piece of code that manage the buzzer sensor:

LCD CODE PART: ***Download***

BUZZER CODE PART: ***Download***

WARNING.- The multisensor board includes an LDR sensor. However, don't use it in case you use BlueTooth or Wi-Fi communications.