

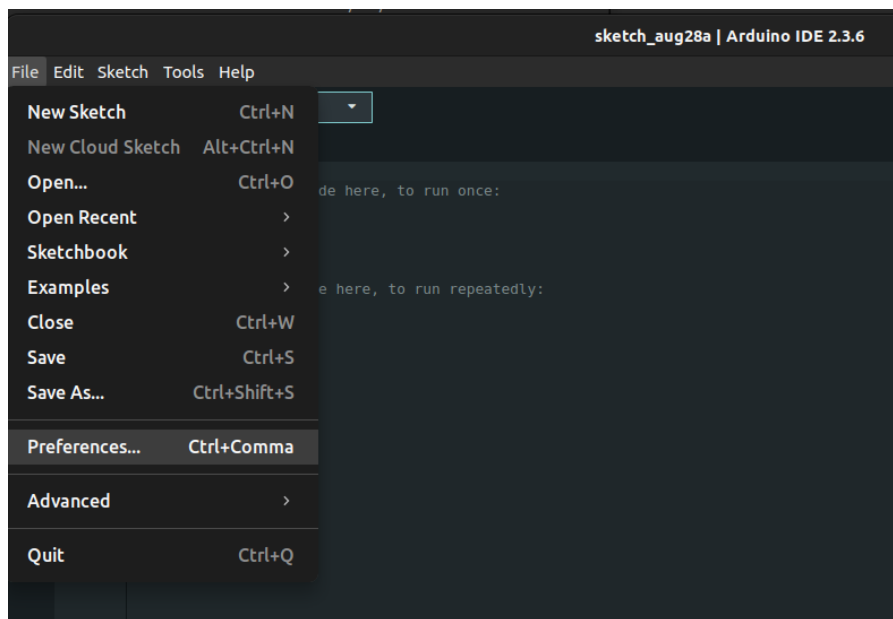
# Lab Session 1: Microcontrollers programming

**Estimated time:** 1.5h (1 session)

## 1. Prepare the Arduino IDE

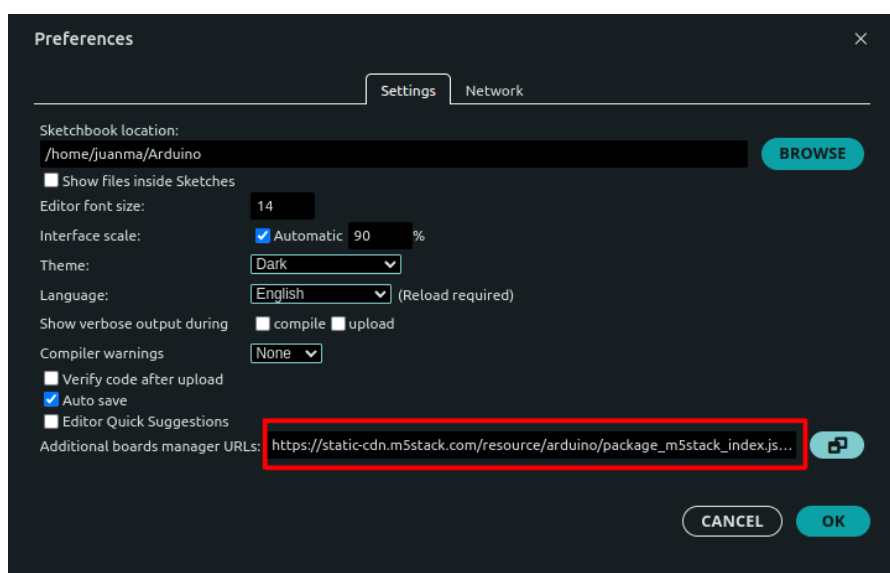
Follow the installation steps described in this guide.

1. Install Arduino IDE: Download it from [this link](#).
2. Install the M5Stack board family:

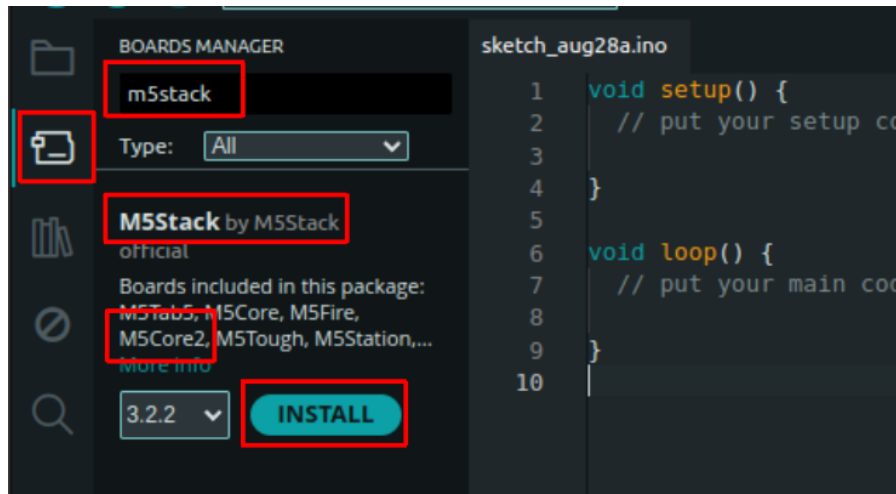


You need to copy/paste this text to download the board package

```
https://static-cdn.m5stack.com/resource/arduino/package_m5stack_index.json
```



Once the board package address is set, you can install it from the *boards manager*.



!!! info This process may take a few minutes. Once installed, select M5Core2 from the board selection menu.

3. Install the Arduino library for M5Core2 (instructions are here.)
4. Install the CP2104 driver. Download it from this link (Step 2: USB driver installation).
5. Compile and upload the `hello_world.ino` example as described in the documentation.

!!! tip "Additional resources" Basic functions for M5Core2

Pinout and Important Notes Below is the M5Core2 pinout. The pins marked in red are the ones we will use in the exercises.

!!! warning - Some *pins* on the M5Core2 are preconfigured, so pay attention when connecting external components. - The ESP32 inside the M5Core2 has 3 serial ports: - `Serial1` is reserved for the display (do not use it). - `Serial0` can be configured (pins G3 – RXD0 and G1 – TXD0), but it is reserved for USB connection to the PC. - `Serial2` is free and can be configured (pins G13 – RXD2 and G14 – TXD2) as regular GPIO using `pinMode()`.

## 2. Turn an LED ON and OFF

Connect an LED as follows:

Run the following program:

```
#include <M5Core2.h>
#define LED_PIN 14

void setup() {
  M5.begin(); // Initialize M5Core2
  pinMode(LED_PIN, OUTPUT);
}

void loop() {
  digitalWrite(LED_PIN, HIGH);
  delay(500);
}
```

```
digitalWrite(LED_PIN, LOW);  
delay(500);  
}
```

!!! question Note that you had to include the `#include <M5Core2.h>` and the `M5.begin(); // Initialize M5Core` Why did you have to do that? what are these instructions for?

## 1. Launch an example

- Access the Wokwi simulation environment via [this link](#).
- Select the ESP32 template.



!!! warning Open the ESP32 template from **Starter Templates**, not from **ESP-IDF Templates**. The former is based on the Arduino IDE while the latter is based on the ESP-IDF environment that will not be used in the course.

- Connect an LED to **GPIO 21** with a resistor, as shown in the diagram (rotate - **R** and flip - **P** components if necessary).



- Simulate the following sketch:

```
#define LED_PIN 21  
#define BUTTON_PIN 35  
  
void setup() {  
    pinMode(LED_PIN, OUTPUT);  
    pinMode(BUTTON_PIN, INPUT_PULLUP);  
}  
  
void loop() {  
    digitalWrite(LED_PIN, HIGH);  
    delay(500);  
    digitalWrite(LED_PIN, LOW);  
    delay(500);  
}
```

!!! question - What is the purpose of the `setup()` function? - What is the purpose of the `loop()` function? - What does the `#define` do? - What does the `pinMode()` function do? - What does the `digitalWrite()` function do? - What does the `delay()` function do?

## 2. Control the LED state with a push button

- Implement the circuit shown in the following diagram and simulate a program that:
  - Turns the LED ON when the button is pressed.

- Turns the LED OFF when the button is not pressed.

!!! tip - Use the function `digitalRead()` to get the status of the button. - You may need to use the `if...else` condition.



!!! warning When you place the button, don't forget to deselect the *bounce* option to avoid **bouncing issues**.

!!! question - What information is included in the `diagram.json`? Did it change with respect to the previous exercise?

- Can you do the exercise without using the `if...else` condition? How?

## 3. Additional exercises

### 3.1. Blink

- Using the same circuit as in the previous exercise, write a script that Blinks the LED while the button is pushed.

!!! tip You may need to use the `millis()` function.

### 3.2. Short press vs long press

- Using the same circuit as in the previous exercise, write a script that does the following:
  - Short press (< 500 ms): Toggle LED (if the LED is ON, turns it OFF, and viceversa).
  - Long press (≥ 500 ms): Blink LED.

!!! tip You may need to use the `millis()` function, and the `if...else` condition.