

Laboratory 1: Basic Arduino and M5stack

Objectives

1. Able to use Arduino IDE with the M5Stack development kit in the lab desktop computer.
2. Able to write program to display characters on the M5Stack LCD Display.

Reference: <https://docs.m5stack.com/en/api/core/lcd>

Arduino Platform

There are several platforms available for Internet of Things (IoT) devices. We carefully selected the Arduino platform, because it is ease and inexpensiveness. In particular, “**Arduino is an open-source electronics prototyping platform based on flexible, easy-to-use hardware and software. It's intended for making interactive projects.**”¹

Arduino IDE Software

Arduino Software is available for free from <http://arduino.cc/en/Main/Software>. Given that the Arduino tool is created in Java, the software can run on any platform that supports Java. These platforms include Windows, macOS and Linux. (Illustration 1 shows a screenshot of Arduino software running on macOS). However, the platform is bundled with AVR GCC and libraries. Thus, we have to create Arduino software in C/C++ style. The Arduino software is enhanced with predefined libraries. Writing software for Arduino is much simpler than writing software for AVR C.

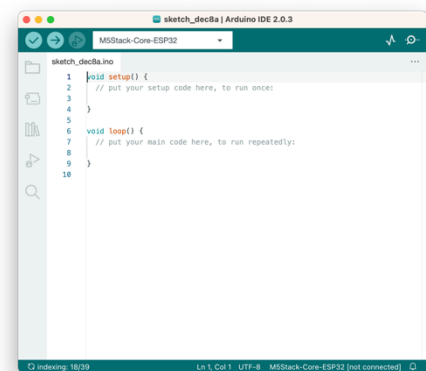


Illustration 1: Screenshot of Arduino Tool

¹The definition is taken from <http://www.arduino.cc>

Arduino substituted the C-style *main* function with *setup* and *loop* functions. The idea is that *setup* only executed once, and *loop* function runs forever. The following is the code to blink LED every seconds.

<pre> /* Blink Turns on an LED on for one second, then off for one second, repeatedly. This example code is in the public domain. */ int led = 13; // the setup routine runs once when you press reset: void setup() { // initialize the digital pin as an output. pinMode(led, OUTPUT); } // the loop routine runs over and over again forever: void loop() { digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level) delay(1000); // wait for a second digitalWrite(led, LOW); // turn the LED off by making the voltage LOW delay(1000); // wait for a second } </pre>	<pre> /* Blink Turns on an LED on for one second, then off for one second, repeatedly. */ #include <avr/io.h> #define F_CPU 16000000UL // X-TAL = 16 MHz #include <util/delay.h> #define LED 5 void delay_ms(unsigned int time) { while(time-->0) { _delay_ms(1.0); } } int main (void) { // initialize the digital pin as an output. DDRB = (1<<LED); // Pin Drive LED = Out // the loop runs over and over again forever: while(1) { PORTB = (1<<LED); // turn the LED on (HIGH is the voltage level) delay_ms(1000); // wait for a second PORTB &= ~(1<<LED); // turn the LED off by making the voltage LOW delay_ms(1000); // wait for a second } } </pre>
(a) Blink in Arduino	(b) Blink in AVR C

Sample code (Blink) in (a) Arduino and (b) AVR C

Sketch

A program is called a sketch² in Arduino (with extension **.ino**). It is the unit of code that is written and uploaded to the Arduino board. Basically, every program written for Arduino is a sketch. It is worth clarifying that only one sketch can be uploaded and run on an Arduino board at a particular time. Uploading a new sketch will replace the existing sketch running on the board.

M5Stack

The Arduino development kit (board) that we will use is M5Stack. M5Stack is a robust, open-source development kit with stackable modules, enabling rapid and high-quality prototyping. The core module contains ESP32, 2.0-inch TFT-LCD 320x240 display, 3 buttons, 1 W speaker, I/O extension, TF-card, i2C bus etc.

²See <http://arduino.cc/en/Tutorial/Sketch> for more details.

Lab Exercises

Save each task into separated sketch, i.e., `lab_1_1.ino`, `lab_1_2.ino`.

Task 1: Set up your Arduino development platform for your M5Stack

1.1 Setup Arduino IDE

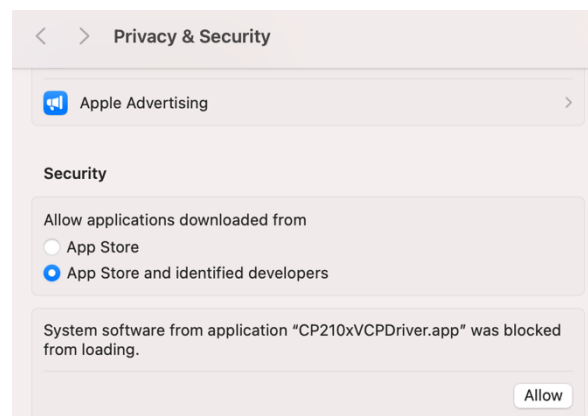
Visit <https://www.arduino.cc/en/software>. Download and install the latest version of **Arduino IDE** for your platform (Windows/MAC/Linux). You need to install into local drive. **DO NOT install to cloud drive, i.e., onedrive, google drive.**

1.2 Setup USB driver

Visit <https://www.silabs.com/developers/usb-to-uart-bridge-vcp-drivers?tab=downloads>. Save **CP210x VCP driver** (zip file) for your platform to a folder. Extract the zip file and run the installer program.

Software • 11		
MacOS	CP210x Universal Windows Driver	v11.3.0 6/24/2023
	CP210x VCP Mac OSX Driver	v6.0.2 10/27/2021
Windows	CP210x VCP Windows	v6.7 9/3/2020
	CP210x Windows Drivers	v6.7.6 9/3/2020
	CP210x Windows Drivers with Serial Enumerator	v6.7.6 9/3/2020
	CP210x_5x_AppNote_Archive	9/3/2020
	CP210x_VCP_Win2K	9/3/2020
	Linux 2.6.x VCP Revision History	9/3/2020

For Mac user, MacOS might block the installation. You need to open **System Settings > Privacy & Security** and click **Allow** to continue installation.

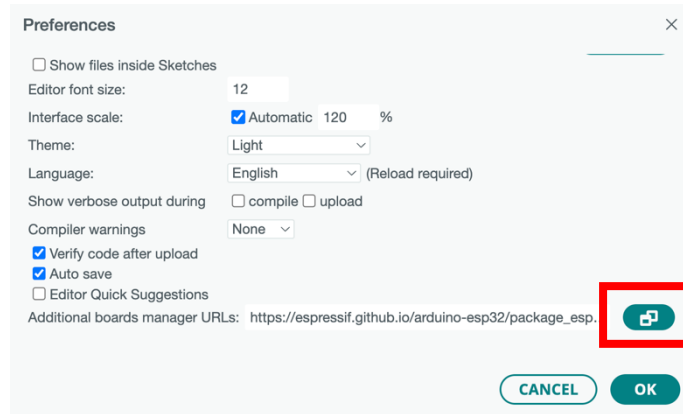



1.3 Setting Arduino IDE Preferences

Open the Arduino IDE. Go to the menu

Mac: *Arduino IDE > Settings...*

Windows: *File > Preferences...*



Scroll down to **Additional Board Manager URLs**;, click on  and insert the URLs

https://espressif.github.io/arduino-esp32/package_esp32_index.json
https://static-cdn.m5stack.com/resource/arduino/package_m5stack_index.json

then click **OK**.

1.4 Setup board and library

1.4.1 Go to the menu **Tools > Boards: > Boards Manager...**, put “m5stack” in the search box and install “M5Stack by M5Stack official”.

1.4.2 Go to the menu **Sketch > Include Library > Manage Libraries...**, put “m5stack” in the search box and install “M5Stack by M5Stack”. Also install all library dependencies.

1.5 Setup USB connection

Connect your M5Stack to your computer (laptop) using the USB cable. In the Arduino IDE, select M5Stack as your board by going to the menu **Tools > Boards > M5Core** and choose the appropriate port in **Tools > PORT** (COM for Windows and UART for Mac) connecting your computer to M5Stack. For Windows users, you can

see the right COM port by inspecting Ports in the device manager. If everything is correct, you should see message:

M5Core on /dev/cu.SLAB_USBtoUART ([Mac](#)), or

M5Core on COM3 ([Windows](#), if connected to COM3 port),

in the status bar at the bottom of Arduino IDE. This will confirm that Arduino IDE can communicate with M5Stack via USB cable. *If this step is not successful, you may need to uninstall and reinstall different versions of the driver (especially for Windows users) until you get to the one that will work for your system.*



1.6 Test with example: HelloWorld

1.6.1 Create your first Arduino sketch by going to the menu **Files > Examples > M5Stack > Basics > HelloWorld**. Alternatively, you can write from scratch by going to the menu **Files > New** and type the following sketch:



```
#include <M5Stack.h>

/* the setup routine runs once when
M5Stack is started or reset */
void setup() {

  M5.begin(); // Initialize the M5Stack
  M5.Lcd.print("Hello World"); // Print text on LCD display
}

/* the loop routine runs over and over again forever */
void loop() {

}
```

1.6.2 Click the **Upload** button  in order to flash the M5Stack. If you just want to verify the sketch without uploading, you can click Verify button .

Troubleshooting: during verify or upload if you get error, read error message.

For Mac OSX users,

a. **Compilation Error:**

Exec: “python”: executable file not found in \$PATH

Solution:

Run the following command in Terminal:

```
sed -i -e 's/=python /=python3 /g' \  
~/Library/Arduino15/packages/esp32/hardware/esp32/*/platform.txt
```

The command modifies all the platform.txt files under your local installation of the ESP32 tools (~/Library/Arduino15/packages/esp32/hardware/esp32) and replaces the word python by python3 wherever it finds it.

<https://forum.arduino.cc/t/exec-python-executable-file-not-found-in-path/971847/16>

b. On Macbook with new M1/M2 CPU, you might error “**Bad CPU type**”. Do the following steps:

1. Open **Terminal**.
2. Type the following command in the **Terminal** window:
`softwareupdate --install-rosetta`
3. Press the **Enter** key.
4. Accept the license agreement with Apple.
5. Wait for the installation to finish.

Task 2: Display text to LCD

Modify the given code to make the LCD screen of M5Stack show your names and ids of all your group members on separate lines with different colors on the LCD screen with text size setting to 5. Use <https://docs.m5stack.com/en/api/core/lcd> as reference.

```
M5.Lcd.setTextSize(textSize) ;
M5.Lcd.setTextColor(textColor) ;
M5.Lcd.setTextColor(textColor, backgroundColor) ;
```

Definition	Hex value	R	G	B
BLACK	0x0000	0	0	0
NAVY	0x000F	0	0	128
DARKGREEN	0x03E0	0	128	0
MAROON	0x7800	128	0	0
PURPLE	0x780F	128	0	128
OLIVE	0x7BE0	128	128	0
LIGHTGREY	0xC618	192	192	192
DARKGREY	0x7BEF	128	128	128
BLUE	0x001F	0	0	255
GREENYELLOW	0xB7E0	180	255	0
GREEN	0x07E0	0	255	0
YELLOW	0xFFE0	255	255	0
ORANGE	0xFDA0	255	180	0
PINK	0xFC9F	255	255	16
CYAN	0x07FF	0	255	255
DARKCYAN	0x03EF	0	128	128
RED	0xF800	255	0	0
MAGENTA	0xF81F	255	0	255
WHITE	0xFFFF	255	255	255

Task 3: Write a paragraph explain what have you learned from this laboratory?

Section: _____ Date: _____

Members

Name: _____ Student ID: _____

Name: _____ Student ID: _____

Name: _____ Student ID: _____

2190151 Computer Programming Laboratory.**Laboratory 1: Basic Arduino and M5Stack**

Task		Graded by
1	Setting up your Arduino development platform for your M5Stack and showing “Hello World” on the LCD screen of M5Stack	
2	Showing names and IDs of all group members on separate lines with different colors on LCD screen with text size setting to 5.	

Answer questions

3	What have you learned from this Laboratory?
---	---