

Lab 1: Setup Development Environment

Objectives:

1. Setup a GitHub repository and a local Git tool.
2. Set up the development environment of Arduino IDE for ESP32-S3.
3. Compile and flash a Blink RGB LED program and modify the LED blinking pattern.

Materials:

1. ESP32-S3 devkit board.
2. Breadboard and Dupont wires.
3. A USB 2.0 cable (Standard-A to Micro-B).
4. Computer

Lab activities:

Task 1

- a. Download and install Git from <https://git-scm.com/download/win> on your computer for local version control. Create a local repository (repo) on your computer using the command line instructions.
- b. Create a readme.txt with the content being your name in one row and commit the file using the command line instructions.

- c. Register a GitHub account for this course and create an online repo on GitHub. You need an email account for this task. Please use this account to share your project work with your team member and the course instructor later in this course.
- d. Push your readme.txt file to your GitHub repo and edit it by adding your matric number in another row. Commit the changes on your GitHub repo.
- e. Clone the GitHub repo <https://github.com/zhjw2023/EE4216> to your local repo for Task 2.
- f. Invite the student next to you to edit your readme.txt to change your name to add his/her name and matric. Commit the changes.

Task 2

- a. Setup the Arduino IDE and download Arduino-ESP32 stable support package for ESP32. You also can refer to the link below.
<https://docs.espressif.com/projects/arduino-esp32/en/latest/installing.html>
- b. Download the Arduino IDE from <https://www.arduino.cc/en/software> and install it with the default settings.
- c. Launch the Arduino IDE. Choose File > Preferences as shown in Figure 1 below. Fill in the following link in the area highlighted in the popup window under the Settings tab and click OK.
https://espressif.github.io/arduino-esp32/package_esp32_index.json

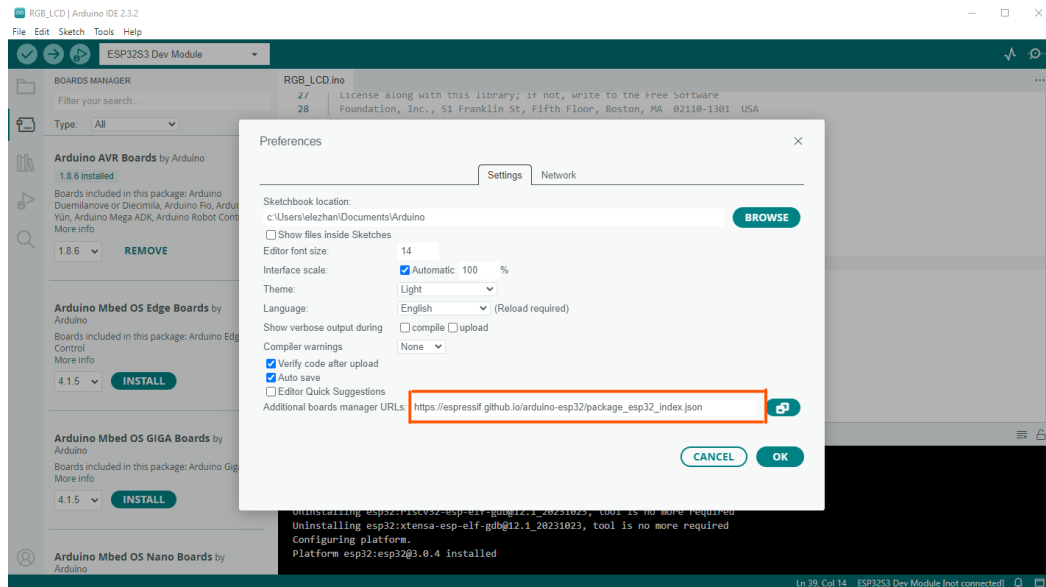


Figure 1

- d. Wait for Arduino-ESP32 support package to be installed. It may take a while. And you need to relaunch the Arduino IDE after the installation finished.
- e. Install the ESP32 board supporting. Select Tools > Board > Board Manager as Figure 2 below shows.

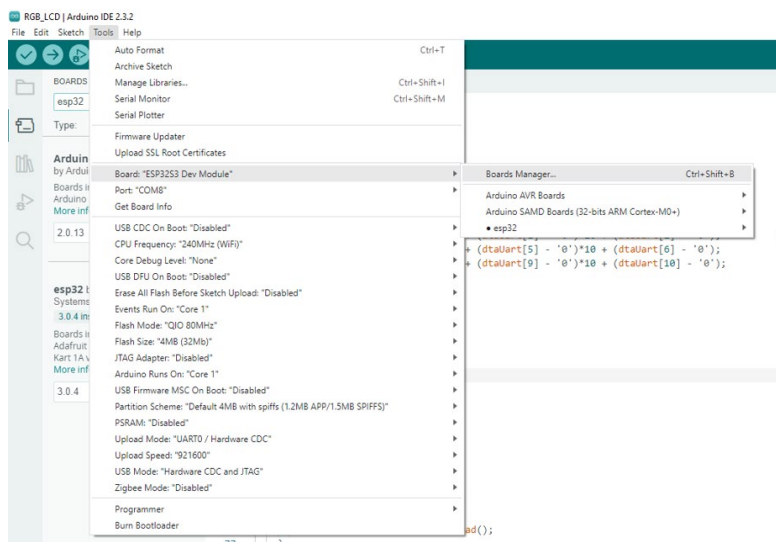


Figure 2

- f. Type “ESP32” in the popup Board Manager window. Choose and install the highlighted package from Espressif Systems shown in Figure 3 below.

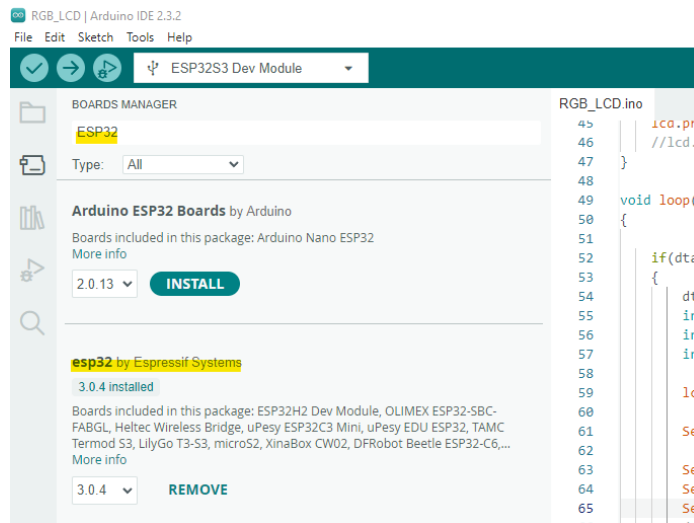


Figure 3

- g. Connect the micro-B side of the USB cable to the USB-to-UART Port on the ESP-32-S3 board as shown by the red circle in Figure 4 below.

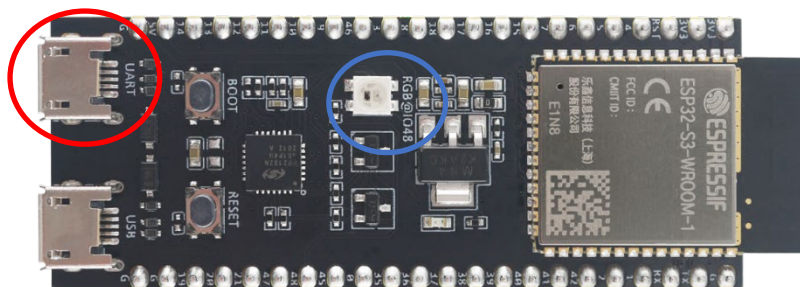


Figure 4

- h. Connect USB A to your computer USB port and wait for the USB port connection notice to popup.

- i. Launch the Device Manager in your windows by typing "device manager" in windows quick-launch window and press enter. Check if you can see the Silicon Lab CP210x USB to UART Bridge listed under Ports (COM& LPT) as shown in Figure 5 below. If it already appears on your system, you can skip Step k below. Otherwise, you need to install the driver as indicated in Step k.

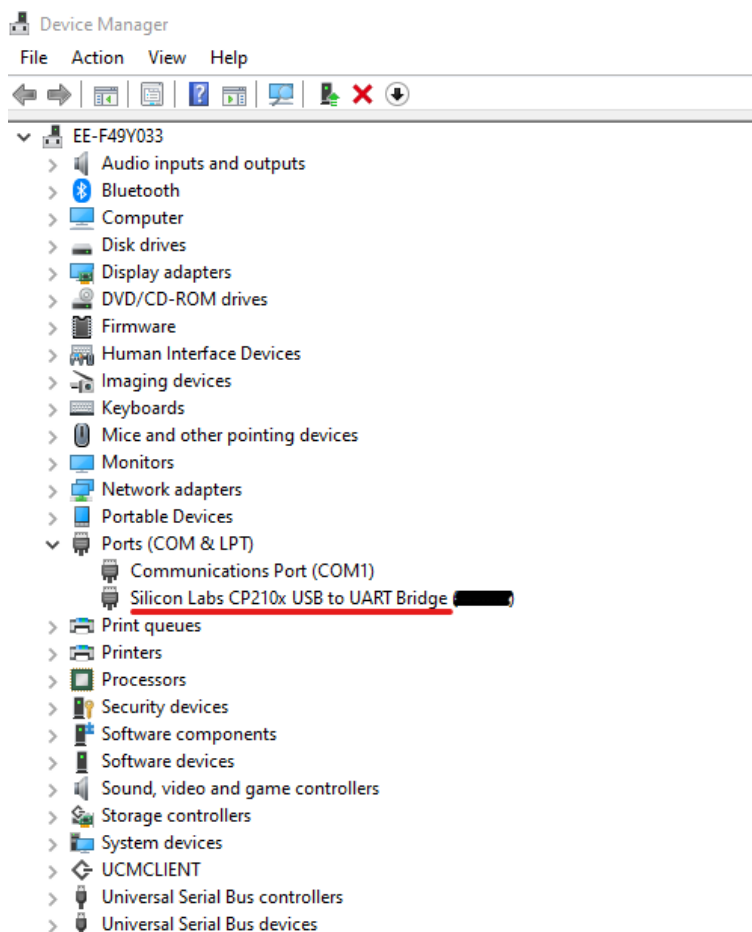


Figure 5

- j. Install driver for the USB-to-UART bridge. You can find the driver program from the cloned folder in Task 1.

- k. Relaunch the Arduino IDE again. And load BlinkRGB.ino, which you have cloned to your local repo in Task 1, through File > Open.
- l. Configure Arduino IDE for BlinkRGB. Select Tools > Board > esp32 > ESP32S3 Dev Module. And Tools > Port and choose your COM port connected to your board. You can find it with the Device Manager if you do not know which COM port it is.
- m. Install the Adafruit NeoPixel RGB LED package for BlinkRGB. As showed in Figure 6.a below, choose Sketch > Include Library > Manage Libraries. In the popup Library Manager window, type “Adafruit NeoPixel” and install the package as highlighted in Figure 6.b.

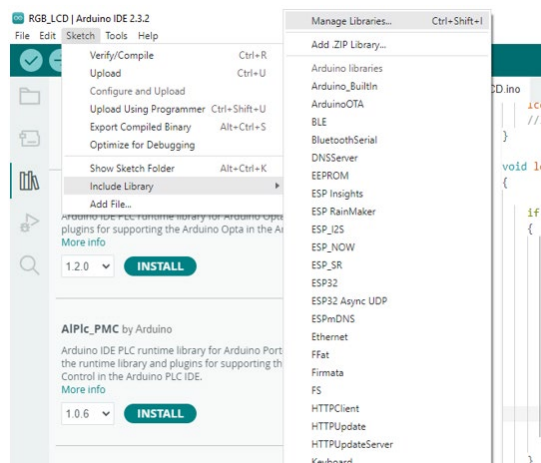


Figure 6.a

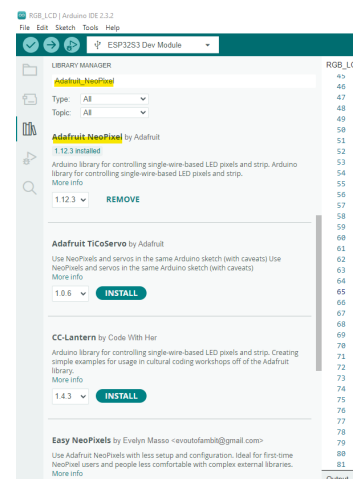


Figure 6.b

- n. Compile and flash the code to your board. And you will see the on board RGB LED is repeating blinking cycle in white color, and in red, green, blue color in a sequence. This shows your environment setup is successful.
- o. Modify the code to show sandy brown, orange red and green each for 1 second and in cycle of 6 seconds. Refer to the color code chart link.

https://www.rapidtables.com/web/color/RGB_Color.html

Assessment: Show your results of the last two steps(Step **n** and Step **o**) to the instructor before the end of your lab session and submit your code in your lab report by 23:59 midnight of your lab session day to Canvas submission folder.