

ReviGrid User Manual Draft

2024 Revidyne

1 Quick Start	3
1.1 ReviGrid Hardware Setup	3
1.2 Uploading an Arduino Sketch.....	4
1.3 Operating Modules Through the Arduino IDE	12

1 Quick Start

This Chapter introduces how to set up the ReviGrid modules and test them using the Arduino IDE for the first time.

1.1 ReviGrid Hardware Setup

Step 1: Open the case and remove the following items

- **All modules.** The case includes 5 modules: the Fan Module, the House Load Module, the Solar Panel Module, the Wind Turbine Module, and the Power Plant Module. Please handle the modules with care as they are instrumented systems. Take special care with the Wind Turbine and the Solar Panel Module as these are both instrumented and actuated.
- **5 short USB C or USB mini cables.** These USB cables connect the modules with the USB hub of the case.
- **1 USB splitter cable.**
- **1 long USB AB or USB C cable.** This cable connects the case with a Windows PC serial port.
- **1 power cable and an AC power adapter.** This cable connects the case with an AC outlet.

Step 2: Connect the data cables to the PC

- Close and latch the case lid.
- Take the long USB AB or USB C cable. Connect the USB B or USB C head to the case's USB B or USB C port. Connect the other end of the cable to the PC's USB A port.

Step 3: Connect the USB cables to the modules

- Connect the short USB C or USB mini cables to each module.
- Place the modules on the case top.




Step 4: Plug the modules into the case

- Plug each module into the closest USB port on the back of the case. Use the USB splitter cable to attach two of the modules to one port. It is recommended to directly connect the Fan Module to the USB port without using the splitter for a stronger power supply.

Step 5: Connect power to the case

- Take the power cable and the AC power adapter.
- Connect the case's power port to an AC outlet.

Step 6: Use Windows Device Manager to verify the presence of each module

- Open the Windows Device Manager  **Device Manager**.
- Expand the *Ports (COM & LPT)* section   **Ports (COM & LPT)**, check the number of serial ports available. As shown in *Figure 1.1*, when all five modules are plugged into the computer, the device manager should show at least 5 serial ports. If any modules have not appeared as devices, unplug that module and replug it.

- *CH340* is the Arduino USB serial driver. *COM<number>* is the COM port number assigned to the device by the Windows Operating System. The COM port number is useful when registering the module's serial port in the Arduino IDE.

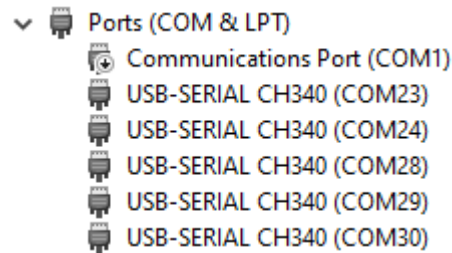


Figure 1.1 Ports (COM & LPT) section in Windows Device Manager

1.2 Uploading an Arduino Sketch

All Arduino sketches are pre-installed on the modules. You may skip this section if you don't want to reupload the sketches. Otherwise, follow the steps to upload the most recent Arduino sketches available for the modules.

Step 1: Download the Arduino IDE



The **Arduino IDE** is a software application used for programming the Arduino microcontrollers. Users can write, compile, upload code to the Arduino boards and enable serial communication with the boards through the Arduino IDE. Check <https://docs.arduino.cc/learn/> to learn more about the Arduino ecosystem.

- Go to <https://www.arduino.cc/en/software> and download the Arduino IDE 2. Choose the applicable download option.

Step 2: Download the Arduino sketches from GitHub

GitHub is a platform that allows developers to edit, store, manage, and share their code. All Arduino sketches for the ReviGrid modules are available on the ReviGrid GitHub Repository.

An **Arduino sketch** is a program (code file) written in the Arduino IDE text editor with the file extension *.ino*.

- Go to https://github.com/PaggieZ/ReviGrid_Public_Repository.git.
- Click on the <> Code icon  and select *Download ZIP*  Download ZIP as shown in *Figure 1.2*.

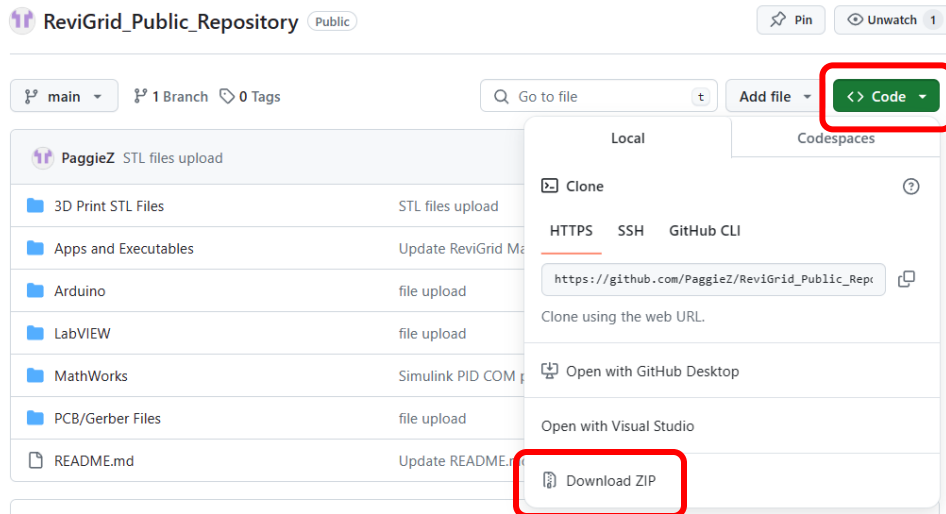


Figure 1.2 Download ZIP icon on GitHub Repository

- Open the zip folder and navigate to the folder in ...\\Arduino\\ReviGrid as shown in Figure 1.3.

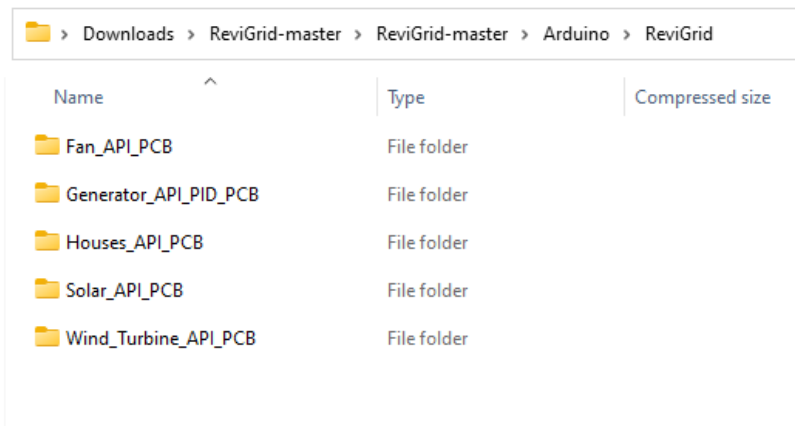


Figure 1.3 ReviGrid folder content

- Open the module folder that you would like to upload. For example, if you want to upload the sketch for the Fan Module, open the *Fan_API_PCB* folder.
- Open the *Fan_API_PCB.ino* file. This will open the Arduino IDE with the Fan Module code as shown in Figure 1.4.

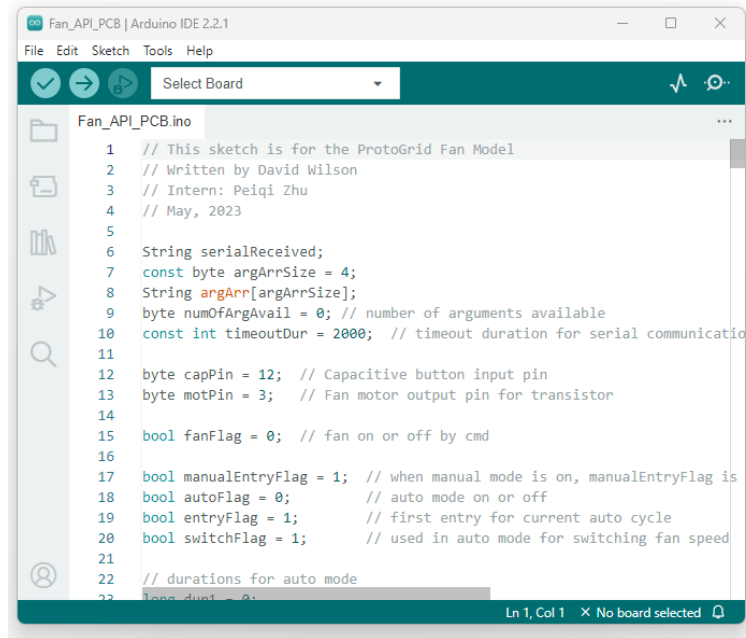




Figure 1.4 Arduino IDE with Fan Module code

Step 3: Configure the Arduino IDE

- Click on the *Select Board* drop down menu  and select the COM port number associated with module. If you are not sure about the COM port number, check the Windows Device Manager  **Device Manager**. You may disconnect all modules and connect only one module to determine the right COM port number.

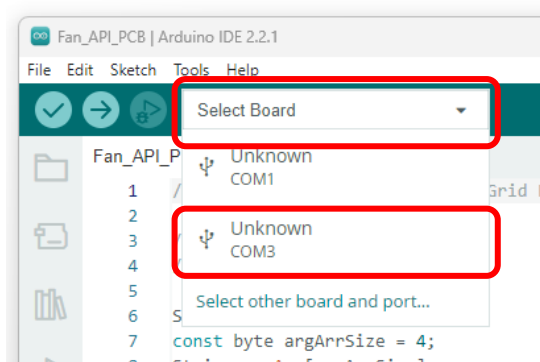


Figure 1.5 Select Board Icon in Arduino IDE

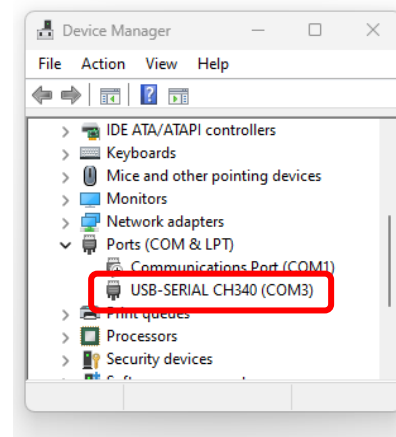


Figure 1.6 COM ports shown in Windows Device Manager

- Select *Arduino Nano* as the board and the corresponding serial port in the pop-up window as shown in *Figure 1.7*. Click *OK* to save the settings.

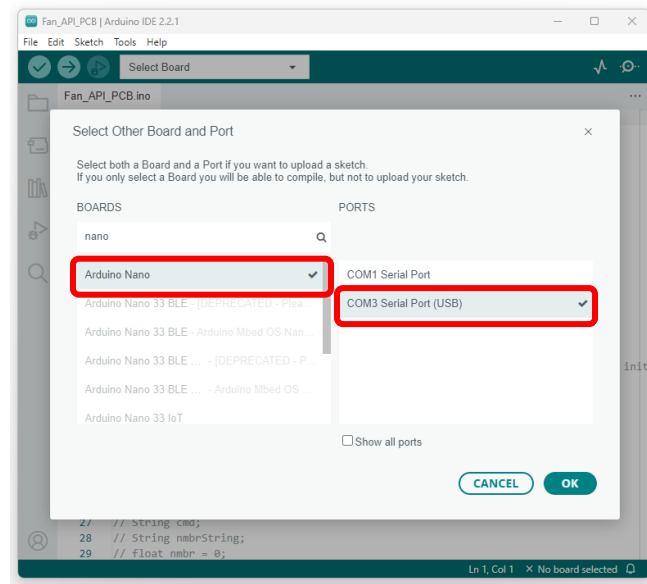


Figure 1.7 Select Other Board and Port Window in Arduino IDE

- Use the shortcut keys **Ctrl+Shift+M** to open the serial monitor. If the module is connected to the Arduino IDE successfully, the serial monitor will display the board name and the COM port number as shown in *Figure 1.8*. Otherwise, it will display a different message as shown in *Figure 1.9*. Try reconnecting the module to resolve the problem. If you have already established serial port connection using another program such as MATLAB or another Arduino IDE window, you will see the message shown in *Figure 1.10*. Simply close the other program to resolve the problem.

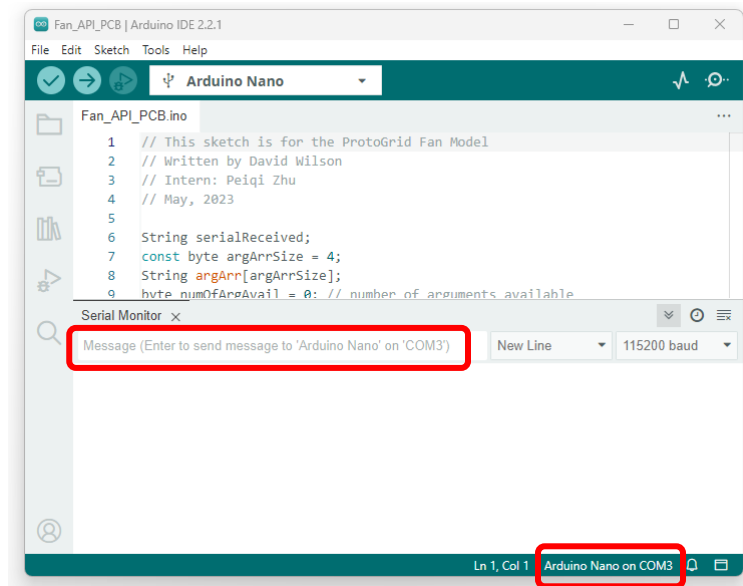


Figure 1.8 Successful serial port connection

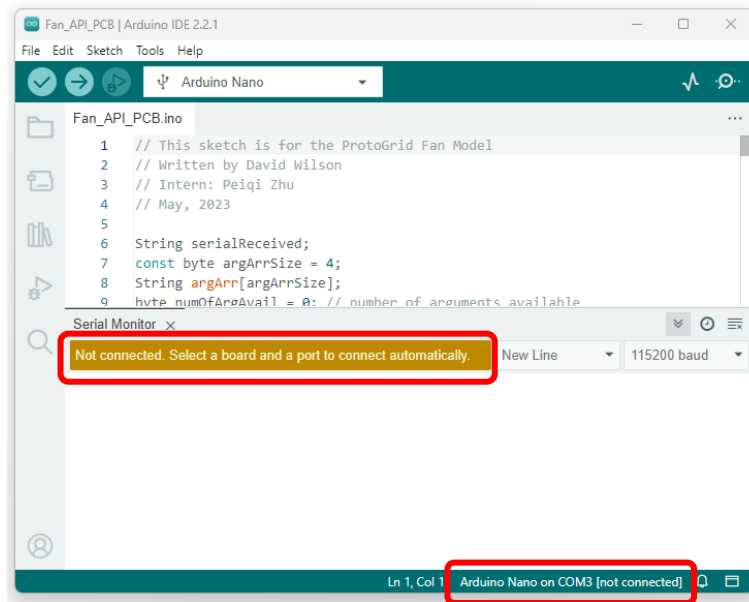


Figure 1.9 Unsuccessful serial port connection

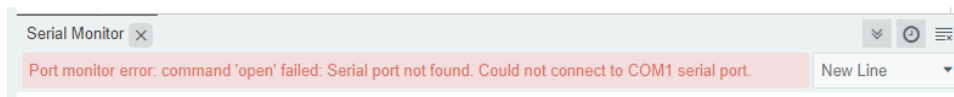


Figure 1.10 Unsuccessful serial port connection due to another program

- Go to *Tools >> Processors* and select “ATmega328P (Old Bootloader)” as shown in Figure 1.11. Some modules require the new bootloader. In that case, select “ATmega328P” as the processor. You will find out the bootloader you need in a later step when uploading the sketch to your module.
- Go to *Tools >> Programmer* and select *Arduino as ISP* as shown in Figure 1.11.

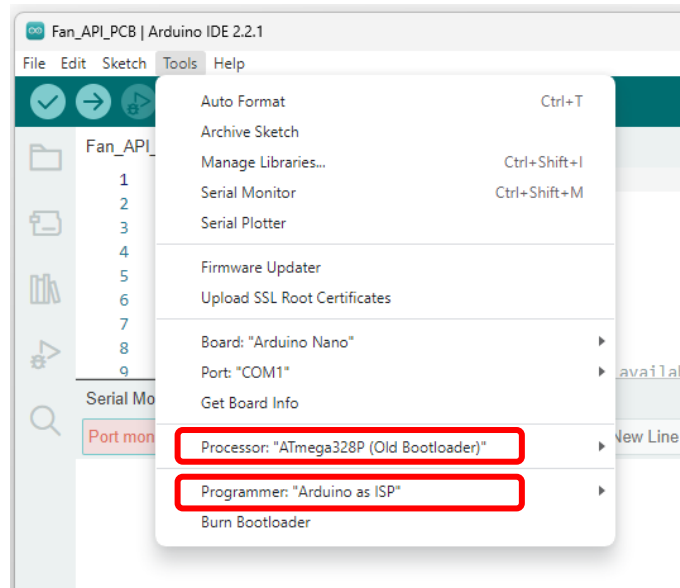



Figure 1.11 Settings for Processor and Programmer

Step 4: Upload the sketch

Click on the *Upload* button  to upload the sketch to the Module. The IDE will show a *Done uploading* message as shown in *Figure 1.12*. If a library is missing as shown in *Figure 1.13*, refer to **Step 5**. If the upload is still unsuccessful, try changing the bootloader through *Tools >> Processors*.

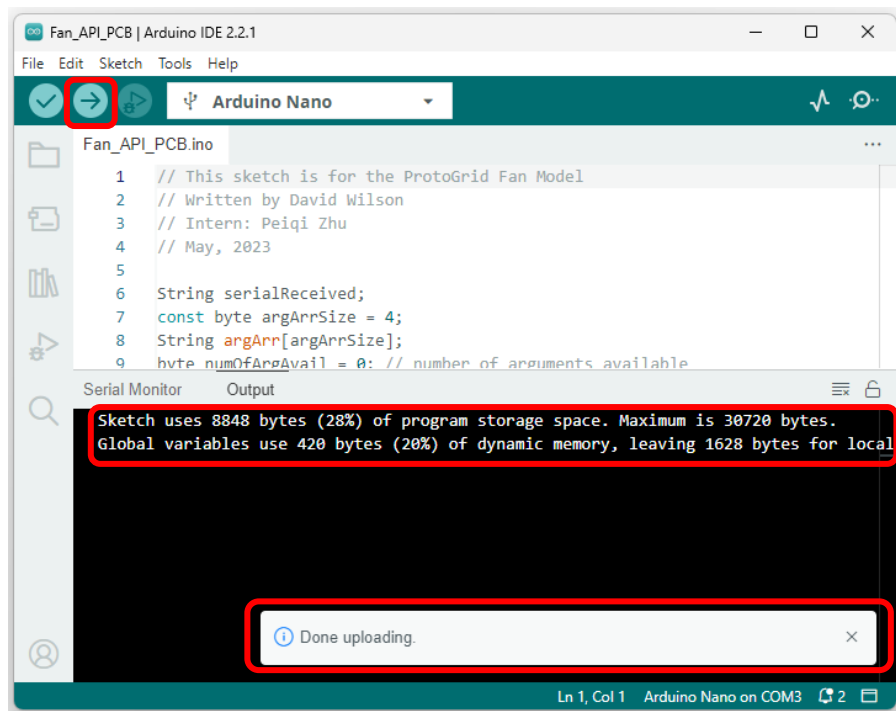


Figure 1.12 Successful upload message

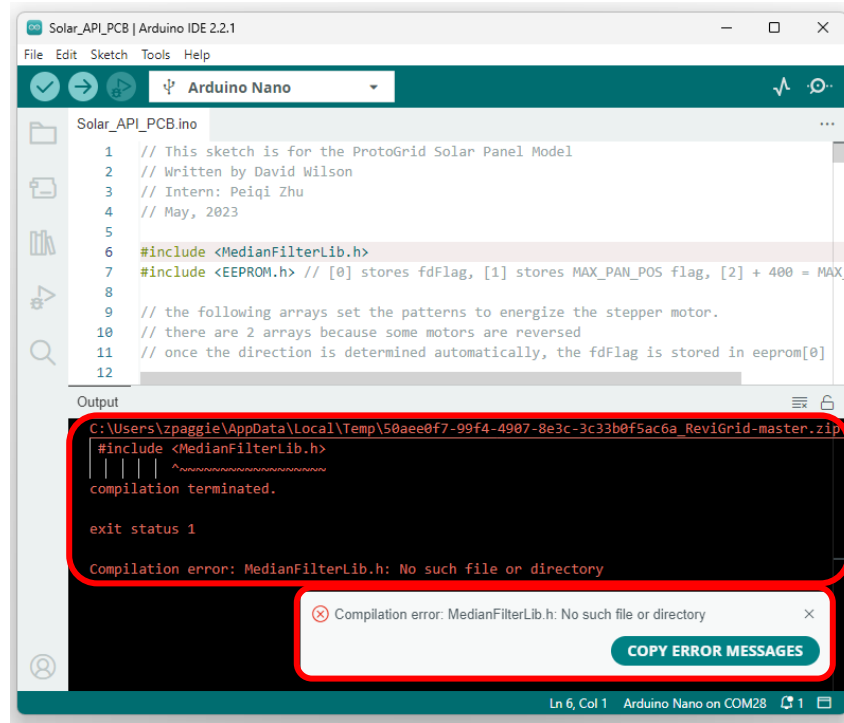



Figure 1.13 Missing library message

Step 5: Download the library

Click on the *Library* button  to find and download the missing libraries. Type in the library name and install the appropriate library. Please refer to *Table 1.1* for library names and authors.

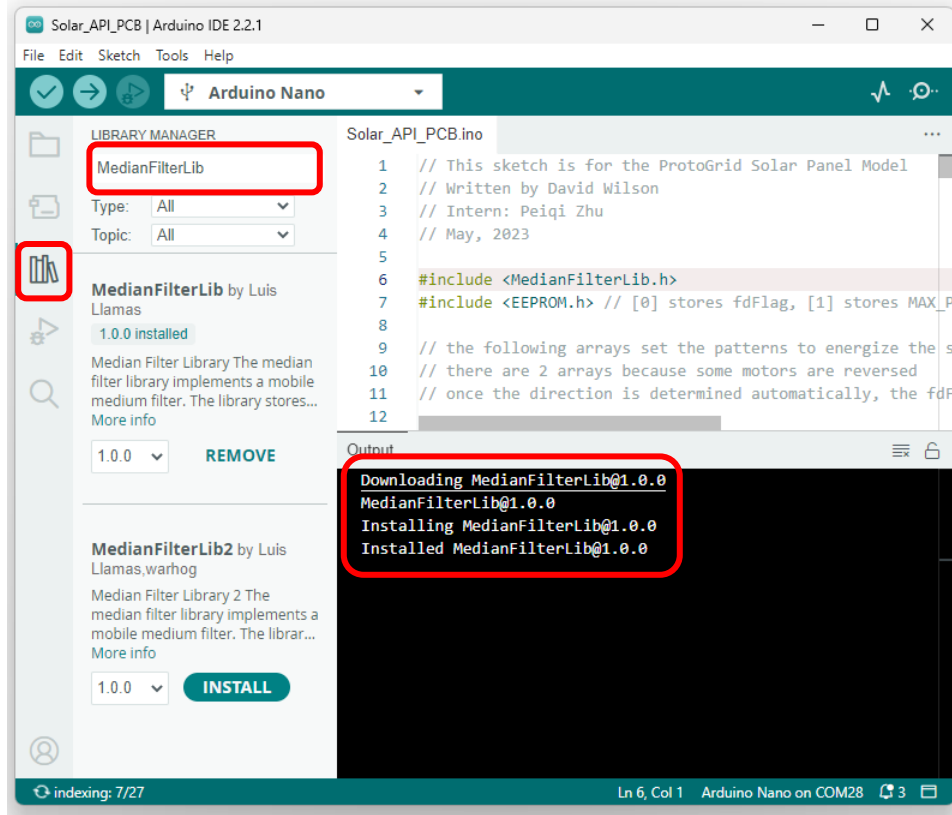


Figure 1.14 Library window

Module	Library	Author
Solar	MedianFilterLib	Luis Llamas
Solar	EEPROM	Built-in
House	Adafruit_sensor	
House	Adafruit_BME280	
Wind Turbine	MedianFilterLib	Luis Llamas
Wind Turbine	EEPROM	Built-in
Power Plant	MedianFilterLib	Luis Llamas
Power Plant	PID_RT	Rob Tillaart
Power Plant	InterpolationLib	Luis Llamas
Power Plant	EEPROM	Built-in

Table 1.1 Library required for each module

1.3 Operating Modules Through the Arduino IDE

In this section, you will use the *Serial Monitor* tool integrated in the Arduino IDE to monitor and control a module by sending string commands and retrieving string feedback through a serial port. Refer to **Step 1** and **Step 3** of the previous section if you have not installed and configured the Arduino IDE.

The **serial communication** protocol allows 2 devices to send and receive serial data through a USB cable. The term “serial” is used because data is sent one bit at a time sequentially. A serial port is a connector on the computer that can transfer data to an external serial device (e.g., an Arduino board). A COM (communication) port number is given to a serial port by the operating system for port identification.

Check <https://docs.arduino.cc/learn/communication/uart>¹ for more information on serial communication.

Step 1: Configure the settings for serial communication

- Launch the Arduino IDE and open the serial monitor (**Ctrl+Shift+M**).
- Configure a serial port by following **Step 3** of the previous section. Since we are only using the Serial Monitor tool in this section, you do not have to open a sketch. If you have plugged in more than one serial device, you may choose any serial port that relates to a ReviGrid module. Check Windows Device Manager for the appropriate serial port to select.
- Set *New Line* as the terminator and 115200 as the baud rate.

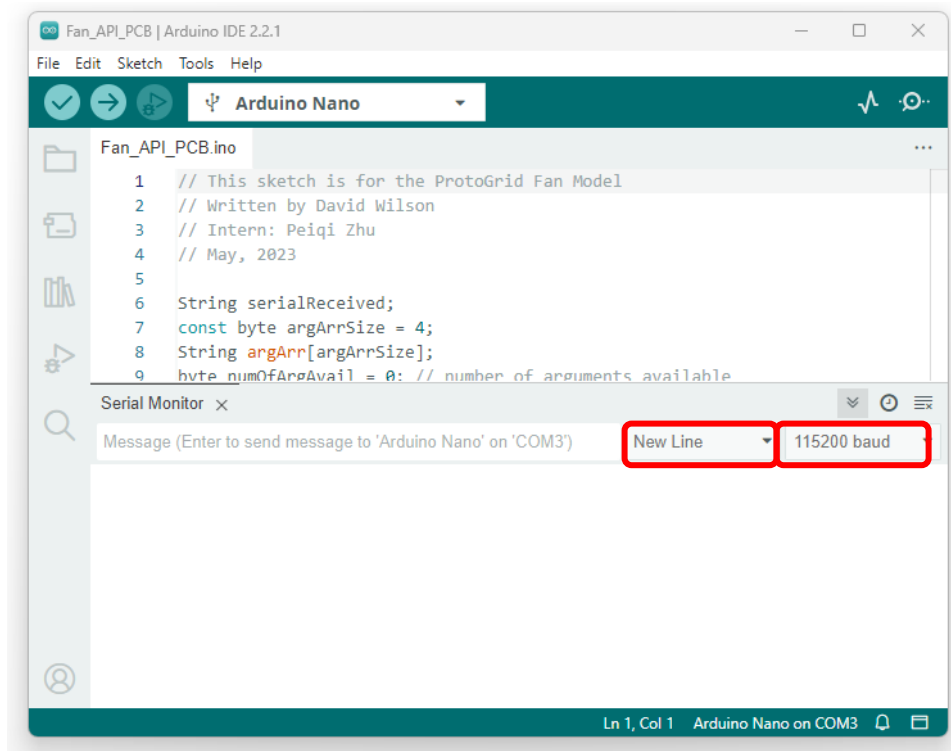



Figure 1.15 Terminator and baud rate selections

¹ Hannes Siebeneicher, “Universal Asynchronous Receiver-Transmitter (UART)”, November 11, 2023, <https://docs.arduino.cc/learn/communication/uart/>

Step 2: Get device ID using **ID?*

- Type **ID?* in the message window then hit *Enter*. The serial monitor should return the device ID that relates to the COM port number you have selected as shown in *Figure 1.16*.
- You can toggle the timestamp by clicking the *Toggle Timestamp* icon .

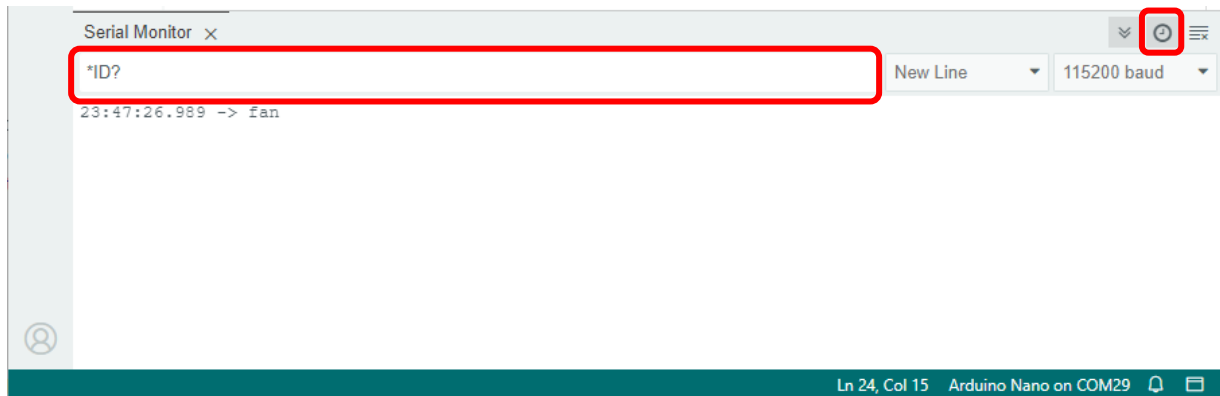


Figure 1.16 Message window location for serial communication

- If you change the serial port and send out the **ID?* again, you will receive a different ID.

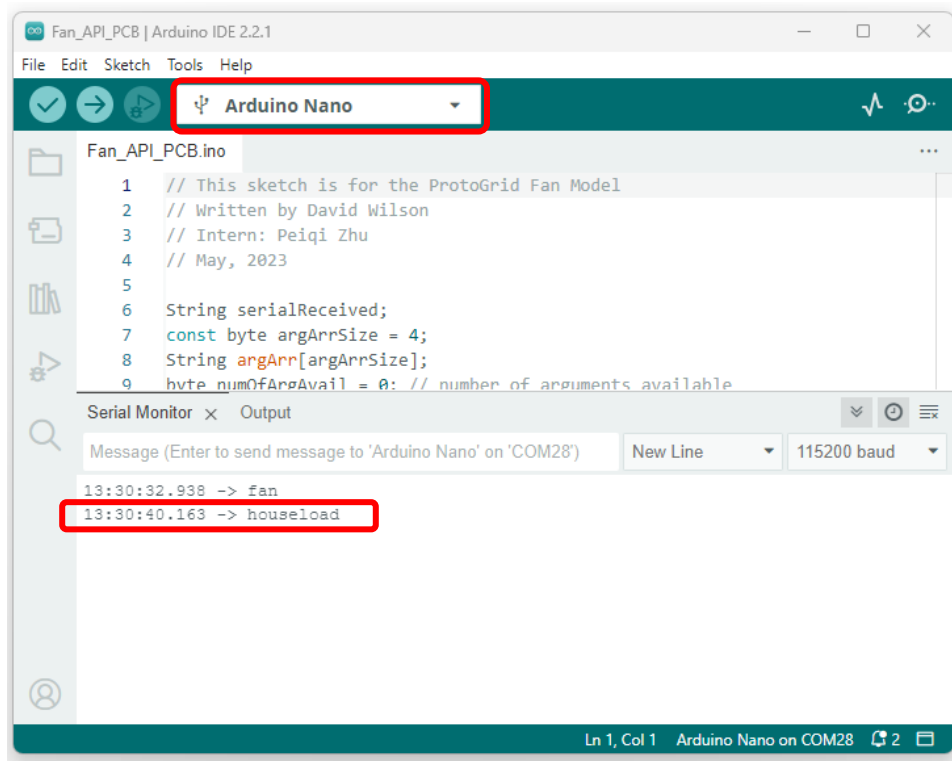


Figure 1.16 Message received when sending **ID?* to the house module

Step 3: Initialize a module using *init*

- Type *init* in the message window then hit *Enter*. The module should respond immediately after you have sent out the message.
- If the module does not respond to the command, try reuploading the Arduino sketch by following the previous section.