

*{Learn, Create, Innovate};*

# Rosserial and Arduino IDE

*Setup*





# Arduino IDE



## Configuring the Arduino IDE

- The Arduino and Arduino IDE are great tools for quickly and easily programming hardware.
- The `rosserial_arduino` package, allows the usage of ROS directly with the Arduino IDE.
- Rosserial provides a ROS communication protocol that works over your Arduino's UART.
- Rosserial, allows Arduino to be a ROS node which can directly publish and subscribe to ROS messages, publish TF transforms, and get the ROS system time.
- Arduino IDE can also be used to program the Hackerboard and other microcontrollers like the ESP32.

A screenshot of the Arduino IDE interface. The top toolbar shows icons for checking, running, saving, and uploading. The file explorer on the left shows a 'Blink' sketch. The main editor area displays the following C++ code:

```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);                     // wait for a second
  digitalWrite(LED_BUILTIN, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);                     // wait for a second
}
```

Arduino IDE



# Arduino IDE



## Installation

- Download the Arduino IDE from the [website](#).
- Install the Arduino IDE application into a folder on the desktop (Windows), or home folder (Ubuntu).
- Follow the installation instructions for Windows and Linux [here](#).
- Once installed, launch the application if you want to select your sketchbook location (File>>Preferences>>Sketchbook location).
  - Sketchbook is a standard place to store your programs, or sketches.
- Close the IDE when done.

Note: The Arduino IDE can be installed in the Virtual Machine, following the same steps.

The top screenshot shows the Arduino IDE 2.0.3 download page. It features the Arduino logo and the text "Arduino IDE 2.0.3". Below this, it states: "The new major release of the Arduino IDE is faster and even more powerful! In addition to a more modern editor and a more responsive interface it features autocompletion, code navigation, and even a live debugger." It also mentions: "For more details, please refer to the [Arduino IDE 2.0 documentation](#)." and "Nightly builds with the latest bugfixes are available through the section below." There is a "SOURCE CODE" section stating: "The Arduino IDE 2.0 is open source and its source code is hosted on [GitHub](#)." To the right, under "DOWNLOAD OPTIONS", there are links for Windows (Win 10 and newer, 64 bits; MSI installer; ZIP file) and Linux (Applimage 64 bits (X86-64); ZIP file 64 bits (X86-64)). Below that are links for macOS (Intel, 10.14: "Mojave" or newer, 64 bits; Apple Silicon, 11: "Big Sur" or newer, 64 bits). An orange arrow points from the "DOWNLOAD OPTIONS" section to a set of symbols on the right, which include a copyright symbol, a registered trademark symbol, a trademark symbol, a 1/4 symbol, and a pound sterling symbol.

The bottom screenshot shows the "Preferences" dialog box in the Arduino IDE. The "Settings" tab is selected. The "Sketchbook location:" field is set to "C:\Users\Mario\Documents\Arduino" and has a "Browse" button next to it. An orange arrow points from the "Sketchbook Location" text to the "Sketchbook location:" field. Other settings include: "Editor language:" set to "System Default" (requires restart of Arduino); "Editor font size:" set to "12"; "Interface scale:" set to "Automatic" (100% requires restart of Arduino); "Theme:" set to "Default theme" (requires restart of Arduino); "Show verbose output during:" checked for "compilation" and "upload"; "Compiler warnings:" set to "None"; "Display line numbers" checked; "Verify code after upload" checked; "Check for updates on startup" checked; "Use accessibility features" checked; "Enable Code Folding" checked; "Use external editor" unchecked; "Save when verifying or uploading" checked. At the bottom, there is a field for "Additional Boards Manager URLs:" with the value "arduino-esp32/gh-pages/package\_esp32\_index.json" and a file icon button. Below this, it says "More preferences can be edited directly in the file C:\Users\Mario\AppData\Local\Arduino15\preferences.txt (edit only when Arduino is not running)".



# Arduino IDE-ESP32 Setup



## Arduino IDE – ESP32 and Hackerboard setup

- To install the ESP32 libraries follow the steps in any of the following links.
  - [Simple tutorial](#) (not official)
  - [Simple tutorial Arduino IDE 2.0](#) (not official)
  - [Espressif Tutorial](#) (official)
  - [Official Github](#)
- Connect the Hackerboard/ESP32 to the computer.
- Test the installation using the example (at the end) [here](#) or [here](#).



# Port Permissions (Ubuntu)

---



- To use Arduino or the ESP32 in Ubuntu, the user must give permissions to the system for accessing ports.
- Make sure the port permissions are granted for the user.
  - In a new terminal type `cd ~/dev` to visualise the port designated by Ubuntu to the MCU. This port are usually called `/ttyACM0` or `/ttyUSB0`.
  - Having obtained the name of the port type the correspondent command to enable the permissions (replace the asterisk with the port number).

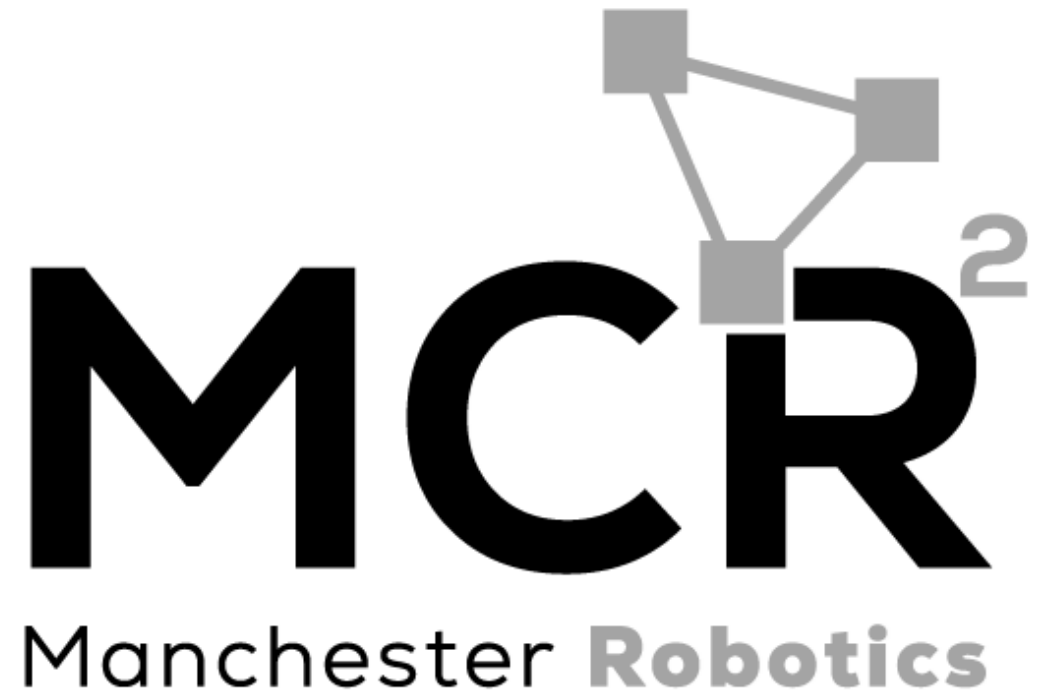
```
sudo chmod 666 /dev/ttyACM*  
sudo chmod 666 /dev/ttyUSB*
```

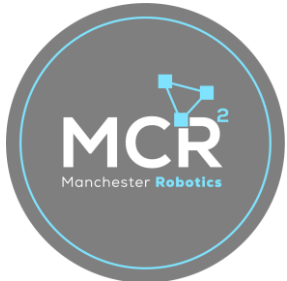
- To make the change permanently, follow the steps [here](#).

# Troubleshoot

*Common problems with  
Arduino IDE*

*{Learn, Create, Innovate};*





# USB Ports in Virtual Machine

---



- When connecting a USB to a VM several steps must be performed for the virtual machine to be able to recognise the USB Port from the host computer. More information can be found [here](#).
- Make sure the correct drivers for the device are installed in the host computer. More information can be found [here](#).
- Give permissions to the VM to access the USB ports of the host machine. More information can be found [here](#) and [here](#).

A video tutorial on how to connect USB devices to the VMWare Player can be found [here](#).



# Troubleshoot (Ubuntu)

---



- When compiling for the ESP32 the following error appears

*“Missing Python: “python”: executable file not found in \$PATH”*

To avoid this error, you can install the python-is-python3 package to create the symbolic links.

```
sudo apt install python-is-python3
```

- When compiling the following error appears

*“ImportError: No module named serial”*

To avoid this error, install the pyserial library

```
sudo apt install python3-pip  
pip3 install pyserial
```

Additional Troubleshoot can be found [here](#), [here](#) and [here](#).



## Troubleshoot (Drivers)

- Drivers are usually installed automatically by Windows and Ubuntu even for the Virtual Machines.
- How do I know if the drivers are properly installed (Windows)?
  - Plug the Puzzle-Bot into the USB port.
  - Go to Start > Device Manager
  - The Serial port should appear as shown in the following figure (The COM port may vary).



- If the computer cannot find the drivers, download the drivers from the following link  
<https://ftdichip.com/drivers/vcp-drivers/>
- Verify that the USB cable is a data cable and not only a power cable!

- Scroll down and download the executable setup as shown in the following figure

Operating System	Release Date	X86 (32-Bit)	X64 (64-Bit)	PPC	ARM	MIPSII	MIPSIV	SH4	Comments
Windows*	2021-07-15	<a href="#">2.12.36.4</a>	<a href="#">2.12.36.4</a>	-	-	-	-	-	WHQL Certified. Includes VCP and D2XX. Available as a <a href="#">setup executable</a>  Please read the <a href="#">Release Notes</a> and <a href="#">Installation Guides</a> .
Linux	-	-	<a href="#">1.5.0</a>	-	-	-	-	-	All FTDI devices now supported in Ubuntu 11.10, kernel 3.0.0-19. Refer to <a href="#">TN-101</a> if you need a custom VCP VID/PID in Linux. VCP drivers are integrated into the <a href="#">kernel</a> .
Mac OS X 10.3 to 10.8	2012-08-10	<a href="#">2.2.18</a>	<a href="#">2.2.18</a>	<a href="#">2.2.18</a>	-	-	-	-	Refer to <a href="#">TN-105</a> if you need a custom VCP VID/PID in MAC OS
Mac OS X 10.9 to 10.14	2019-12-24	-	<a href="#">2.4.4</a>	-	-	-	-	-	This driver is signed by Apple

### Before Installing the drivers!!

- Unplug the Puzzle-Bot from the computer.
- Unzip the drivers and run the setup (some computers are required to be restarted after the installation).
- Plug the Puzzle-Bot back into the computer.



# Troubleshoot

---



## Troubleshoot (Drivers)

- Some Hackerboards have a different USB-UART chip the CP210x.
- Drivers are usually installed automatically by Windows and Ubuntu even for the Virtual Machines.
- Verify if they are installed by following the steps in the previous slide.
- Verify that the USB cable is a data cable and not only a power cable!.
- If the computer cannot find the drivers, download the drivers from the following link

<https://www.silabs.com/developers/usb-to-uart-bridge-vcp-drivers?tab=downloads>

## Before Installing the drivers!!

- Unplug the Puzzle-Bot from the computer.
- Unzip the drivers and run the setup (some computers are required to be restarted after the installation).
- Plug the Puzzle-Bot back into the computer.

A troubleshoot guide can be found [here](#).





# Troubleshoot



## Troubleshoot (Drivers)

- My computer still not recognize the drivers even after the installation
- Plug the Puzzle-Bot into the USB port.
- Go to Start > Device Manager.
- Look for the USB Serial Converter as shown in the following picture.

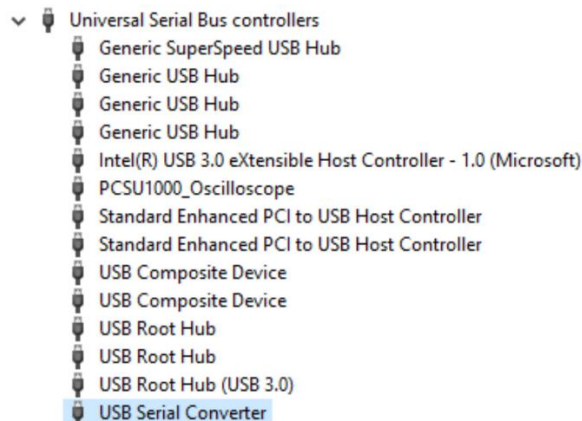


FIGURE: USB SERIAL CONVERTER

- Right Click to Properties > Advanced Tab.
- Make sure the Load VCP box is checked.
- Reconnect the Puzzle-Bot to the computer.

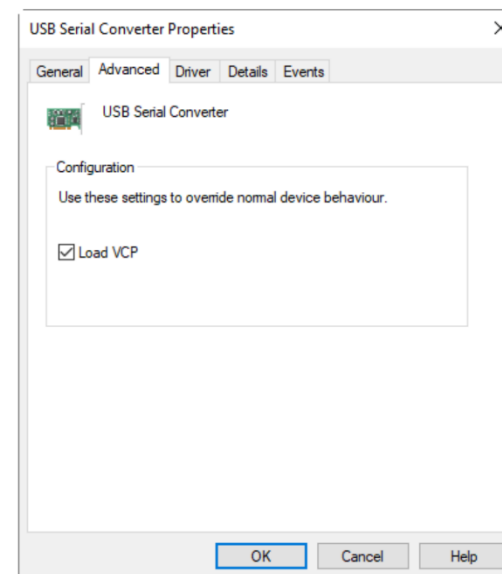


FIGURE: VCP PORT