Higgs 2.0 User & Maintenance Manual

Álvaro Aparicio Serna
Jorge Brihuega Segura
Enrique de Diego Henar
Alejandro Fariñas Nubla
Gonzalo Fernández Moreno
Iñaki Guillén Elizari
David J. Mediavilla Pérez
Alfredo Núñez Martínez
Lorenzo Prat Boubeta
Diego Rodríguez García
Alberto Sánchez Corral
David Sánchez González
Pedro J. Soler Olivares

25/10/2021

ISE FILE – 01

Document Revision List

<table>
<tr><td><b>Document</b></td><td colspan="5">Document Name</td></tr>
<tr><td><b>Project</b></td><td colspan="3">ISE-IC-2021</td><td><b>CI UID</b></td><td>ZZZZ</td></tr>
<tr><td><b>Version</b></td><td><b>Date</b></td><td><b>Author(s)</b></td><td><b>Reviewer(s)</b></td><td colspan="2"><b>Summary of Changes and Comments</b></td></tr>
<tr><td>1.0</td><td>11/06/2022</td><td>Alfredo Nuñez & Lorenzo Prat</td><td></td><td colspan="2">User &amp; Maintenance Manual</td></tr>
<tr><td></td><td></td><td></td><td></td><td colspan="2"></td></tr>
<tr><td></td><td></td><td></td><td></td><td colspan="2"></td></tr>
<tr><td></td><td></td><td></td><td></td><td colspan="2"></td></tr>
</table>

Index

1. Hardware

1.1. Arduino Uno

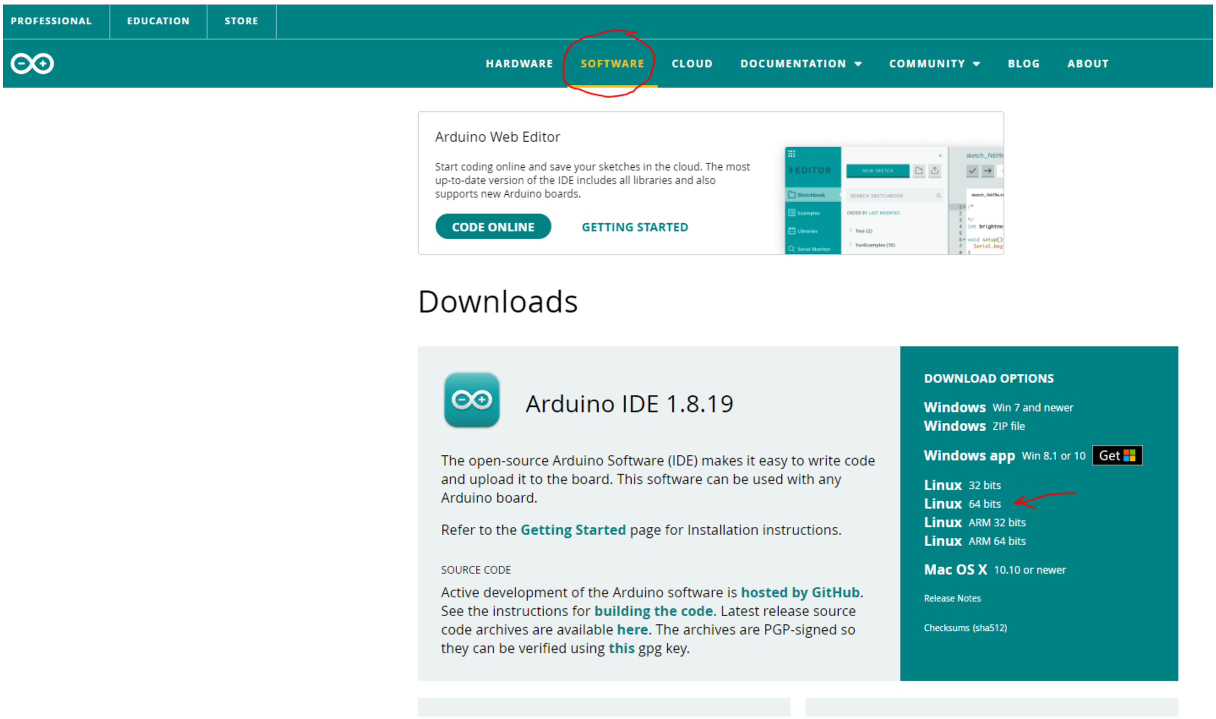
Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards can read inputs and turn them into outputs, that's why they are commonly used for DIY projects.

In this project, we are using the Arduino UNO board, which has: an ATmega328P processor, 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog input download crystal, a USB connection, a power jack, and an ICSP header and a reset button.

To program these boards you have multiple options. In this document, the two most used will be presented.

1. Working with Arduino IDE

- To download the Arduino IDE the user must do it from the original website as can be seen in Fig. 1. 1: https://www.arduino.cc/en/software.



Arduino Web Editor

Start coding online and save your sketches in the cloud. The most up-to-date version of the IDE includes all libraries and also supports new Arduino boards.

Downloads

Arduino IDE 1.8.19

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Refer to the **Getting Started** page for Installation instructions.

SOURCE CODE

Active development of the Arduino software is **hosted by GitHub**. See the instructions for **building the code**. Latest release source code archives are available **here**. The archives are PGP-signed so they can be verified using **this** gpg key.

DOWNLOAD OPTIONS

Windows Win 7 and newer
Windows ZIP file

Windows app Win 8.1 or 10 Get

Linux 32 bits
Linux 64 bits
Linux ARM 32 bits
Linux ARM 64 bits

Mac OS X 10.10 or newer

Release Notes

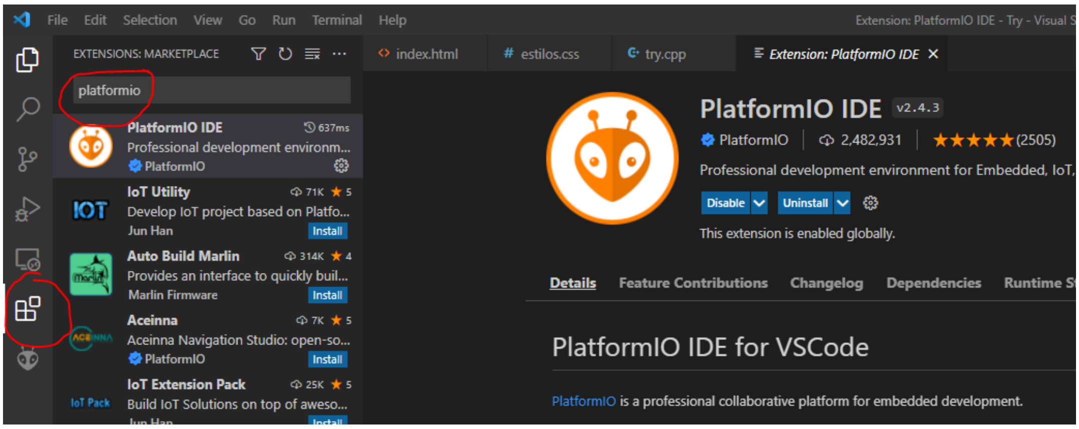
Checksums (sha512)

Fig. 1. 1 Arduino IDE download page

- After the installation, the Arduino IDE will be ready to use and it has multiple examples to get started with this working environment. Also, tutorials can be found by searching the internet or on the same original web for the download of the program.

2. Working with PlatformIO (Virtual Studio Code)

- If the user is used to working in Virtual Studio Code for its projects, the recommended working environment is PlatformIO, which can be downloaded as an extension from the Virtual Studio Code environment, as shown below:



File Edit Selection View Go Run Terminal Help
Extension: PlatformIO IDE - Try - Visual S
EXTENSIONS: MARKETPLACE
index.html
estilos.css
try.cpp
Extension: PlatformIO IDE
platformio
PlatformIO IDE 637ms
Professional development environm...
PlatformIO
IoT Utility 71K ★ 5
Develop IoT project based on Platfo...
Jun Han Install
Auto Build Marlin 314K ★ 4
Provides an interface to quickly buil...
Marlin Firmware Install
Aceinna 7K ★ 5
Aceinna Navigation Studio: open-so...
PlatformIO Install
IoT Extension Pack 25K ★ 5
Build IoT Solutions on top of aweso...
Jun Han Install
PlatformIO IDE v2.4.3
PlatformIO 2,482,931 ★★★★★(2505)
Disable Uninstall
Details Feature Contributions Changelog Dependencies Runtime S

Professional development environment for Embedded, IoT,

This extension is enabled globally.

PlatformIO IDE for VSCode

PlatformIO is a professional collaborative platform for embedded development.

- This extension allows the user to program and upload code to the Arduino board as the Arduino IDE does, and, in addition, it provides the location of the errors in the code. The problem is that this extension does not allow to see the Serial Prompt that the Arduino IDE has, but the user can combine both working environments.

1.2. Motor Driver

For powering the motors of the robot, two drivers L298N connected in parallel are used, each one to power two motors at the same time.

This driver is powered by 12V from the batteries but it admits from 3V to 35V. Also, it provides 2-3A as an output in each channel, which is enough to power the motors.

To learn more about how this driver works, the recommended tutorial is the following: https://www.luisllamas.es/arduino-motor-corriente-continua-l298n/. Moreover, to see the connections between the boards and the batteries will be displayed later in this document.

1.3. Nvidia Jetson Xavier NX & Auvidea JNX30D

1. Before you start

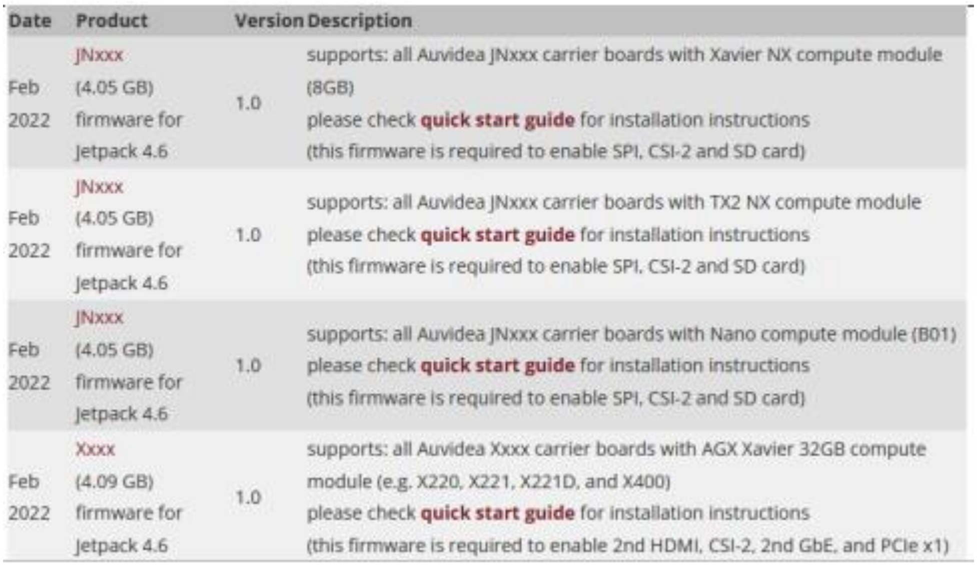
- Please make sure to use a Linux host PC with Ubuntu 18.04 LTS or Ubuntu 20.04 LTS (other versions may work but some caused problems in the past!) operating system. Please use a native setup (no virtual machine). This Host PC should have a high bandwidth internet connection for the download of the 3GByte+ installation file in the following steps.

- You will also need a high-quality standard USB 2.0 Type-A to micro-USB 2.0 cable.

2. Download the installation file from Auvidea

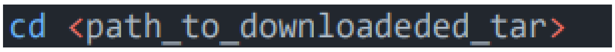
1. Download the installation file for your setup from Auvidea. JNxxx means the carrier board series starting with JN, for example, JN30D. Please check the description to download the right file for your module Xavier NX

https://auvidea.eu/firmware/



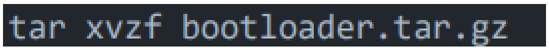
Date	Product	Version	Description
Feb 2022	JNxxx (4.05 GB) firmware for Jetpack 4.6	1.0	supports: all Auvidea JNxxx carrier boards with Xavier NX compute module (8GB) please check **quick start guide** for installation instructions (this firmware is required to enable SPI, CSI-2 and SD card)
Feb 2022	JNxxx (4.05 GB) firmware for Jetpack 4.6	1.0	supports: all Auvidea JNxxx carrier boards with TX2 NX compute module please check **quick start guide** for installation instructions (this firmware is required to enable SPI, CSI-2 and SD card)
Feb 2022	JNxxx (4.05 GB) firmware for Jetpack 4.6	1.0	supports: all Auvidea JNxxx carrier boards with Nano compute module (B01) please check **quick start guide** for installation instructions (this firmware is required to enable SPI, CSI-2 and SD card)
Feb 2022	Xxxx (4.09 GB) firmware for Jetpack 4.6	1.0	supports: all Auvidea Xxxx carrier boards with AGX Xavier 32GB compute module (e.g. X220, X221, X221D, and X400) please check **quick start guide** for installation instructions (this firmware is required to enable 2nd HDMI, CSI-2, 2nd GbE, and PCIe x1)

2. Open a terminal window (CTRL + ALT + T) on your Linux host PC and navigate to your download location.



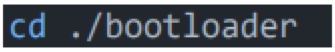
```
cd <path_to_downloadeded_tar>
```

3. Extract the tar.gz file you just downloaded.



```
tar xvzf bootloader.tar.gz
```

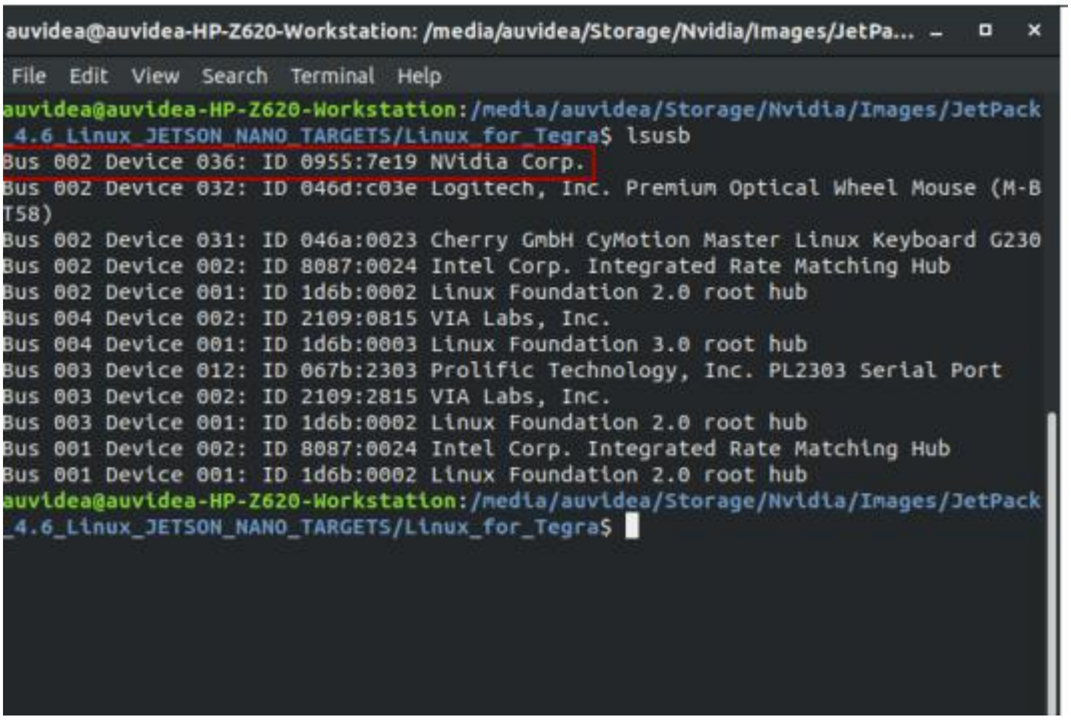
4. Change the directory to the extracted bootloader folder



```
cd ./bootloader
```

3. Connect carrier board to host PC

1. Connect the system to the Linux host PC. Please use a USB 2.0 cable (micro-USB on the carrier board).
2. After connecting to the host PC power up the system. The system will detect the host PC and automatically enter the flashing state (also called force recovery mode).
3. Check that the connection is established with the lsusb command. You should find one entry with Nvidia Corp. as highlighted below.

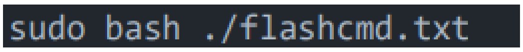


auvidea@auvidea-HP-Z620-Workstation: /media/auvidea/Storage/Nvidia/Images/JetPa...

```
auvidea@auvidea-HP-Z620-Workstation:/media/auvidea/Storage/Nvidia/Images/JetPack
_4.6_Linux_JETSON_NANO_TARGETS/Linux_for_Tegra$ lsusb
Bus 002 Device 036: ID 0955:7e19 NVidia Corp.
Bus 002 Device 032: ID 046d:c03e Logitech, Inc. Premium Optical Wheel Mouse (M-B
T58)
Bus 002 Device 031: ID 046a:0023 Cherry GmbH CyMotion Master Linux Keyboard G230
Bus 002 Device 002: ID 8087:0024 Intel Corp. Integrated Rate Matching Hub
Bus 002 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 004 Device 002: ID 2109:0815 VIA Labs, Inc.
Bus 004 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 003 Device 012: ID 067b:2303 Prolific Technology, Inc. PL2303 Serial Port
Bus 003 Device 002: ID 2109:2815 VIA Labs, Inc.
Bus 003 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 001 Device 002: ID 8087:0024 Intel Corp. Integrated Rate Matching Hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
auvidea@auvidea-HP-Z620-Workstation:/media/auvidea/Storage/Nvidia/Images/JetPack
_4.6_Linux_JETSON_NANO_TARGETS/Linux_for_Tegra$
```

4. Flashing of system

1. Use the flashcmd script in the extracted bootloader folder to transfer the software into the Jetson compute module and flash it.



```
sudo bash ./flashcmd.txt
```

2. Please connect a monitor to the system. After the flashing process has completed the should automatically boot and show the Ubuntu desktop.

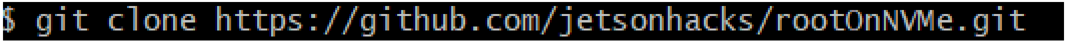
5. Root from SSD

This guide describes how to set up your system to boot fully from an M.2 SSD

Switch the rootfs to an NVMe SSD on the Jetson Xavier NX and Jetson AGX Xavier

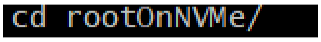
These scripts install a service that runs at startup to point the rootfs to an SSD installed on /dev/nvme0 (the M.2 Key M slot).

1. This procedure should be done on a fresh install of the SD card using JetPack 4.3+. Install the SSD into the M.2 Key M slot of the Jetson, format it gpt, ext4, and set up a partition (p1).
2. Clone the GitHub repository jetsonhacks/**rootOnNVMe**



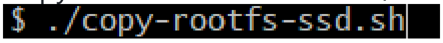
```
$ git clone https://github.com/jetsonhacks/rootOnNVMe.git
```

3. Change to de **rootOnNVMe** file.



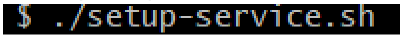
```
cd rootOnNVMe/
```

4. Copy the rootfs of the eMMC/SD card to the SSD



```
$ ./copy-rootfs-ssd.sh
```

5. Setup the service. This will copy the. service file to the correct location, and install a startup script to set the rootfs to the SSD.

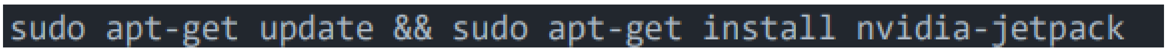


```
$ ./setup-service.sh
```

6. After setting up the service, reboot for the changes to take effect.

6. Installing additional NVIDIA SDK components

1. Now you can install additional NVIDIA SDK components. Please connect the system to the internet. Open a terminal window on the system (CTRL ALT T). Use apt-get to install the components. If this fails, please check the Internet connection of the system.

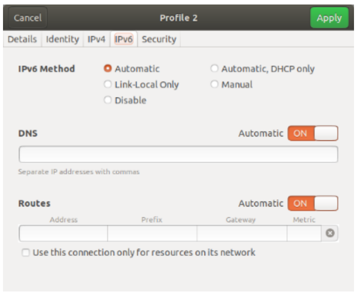


```
sudo apt-get update && sudo apt-get install nvidia-jetpack
```

7. Connect to Nvidea Jetson vias SSH

At this point, we will explain the steps needed to connect the Nvidea Jetson with a host PC.

1. Connect both devices via ethernet cable.
2. Select the 192.100.100.0 network at the Nvidea Jetson (default setup).
3. On the host computer create a new network wired profile, with the following details.



Profile 2 — IPv6

IPv6 Method: ● Automatic ○ Automatic, DHCP only ○ Link-Local Only ○ Manual ○ Disable

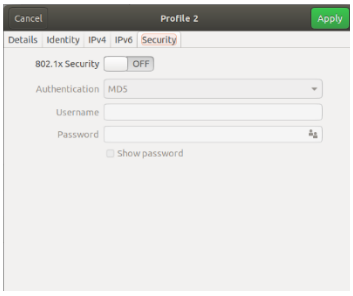
DNS — Automatic: ON

Separate IP addresses with commas

Routes — Automatic: ON

Address	Prefix	Gateway	Metric

□ Use this connection only for resources on its network



Profile 2 — Security

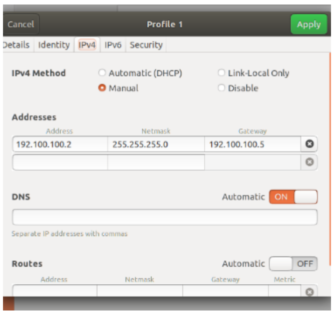
802.1x Security: OFF

Authentication: MD5

Username:

Password:

□ Show password



Profile 1 — IPv4

IPv4 Method: ○ Automatic (DHCP) ○ Link-Local Only ● Manual ○ Disable

Addresses

Address	Netmask	Gateway
192.100.100.2	255.255.255.0	192.100.100.5

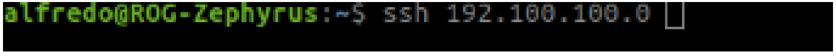
DNS — Automatic: ON

Separate IP addresses with commas

Routes — Automatic: OFF

Address	Netmask	Gateway	Metric

4. Open a terminal and write the following the command



```
alfredo@ROG-Zephyrus:~$ ssh 192.100.100.0
```

5. Introduce the password of the user at the Nvidea Jetson.
6. After following all these steps, the terminal at your host computer will have transformed into a terminal from the Nvidea jetson

2. Software

The Arduino Uno will be communicated with the Nvidea Jetson which will be the main Processor Unit of the Higgs 2.0. For these, a Ros-Arduino library has been installed at the Nvidea Jetson.

Nvidia Jetson

The Board is running on Ubuntu 18.04 and Ros melodic, which has been installed according to the following tutorial https://github.com/jetsonhacks/rootOnNVMe.

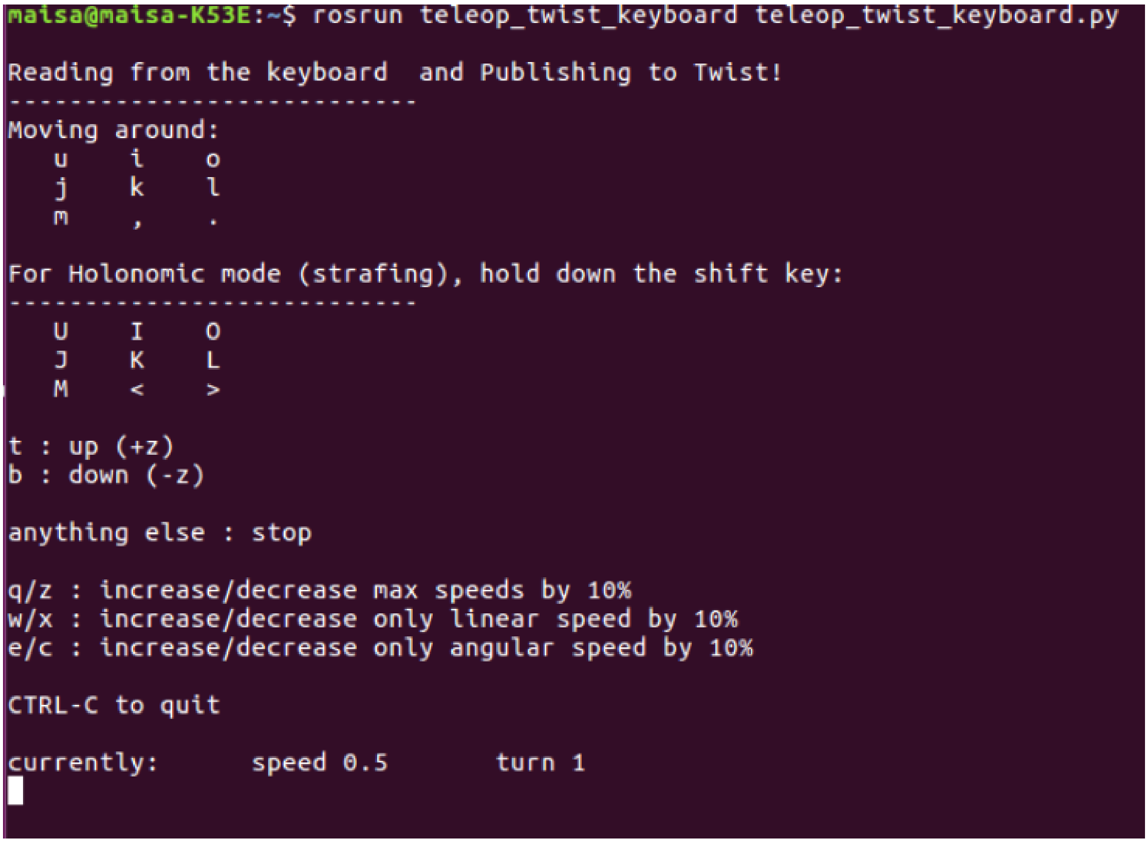
1. Move the Higgs 2.0 Teleoperate

1. To move the Higgs first all the steps from Point 3.7 from the Software section must be followed.
2. Run the Ros Arduino serial package for the communication between the Nvidea Jetson and the Arduino.

```
rosrun rosserial_python serial_node.py /dev/XXXXXXXX
```

Change the XXXXX with the port used by the Arduino.
You can find this information in the Arduino Ide **Tools/port** section**.**

3. Run the Teleop package which already has been installed at the Nvidea Jetson.



```
maisa@maisa-K53E:~$ rosrun teleop_twist_keyboard teleop_twist_keyboard.py

Reading from the keyboard  and Publishing to Twist!
---------------------------
Moving around:
   u    i    o
   j    k    l
   m    ,    .

For Holonomic mode (strafing), hold down the shift key:
---------------------------
   U    I    O
   J    K    L
   M    <    >

t : up (+z)
b : down (-z)

anything else : stop

q/z : increase/decrease max speeds by 10%
w/x : increase/decrease only linear speed by 10%
e/c : increase/decrease only angular speed by 10%

CTRL-C to quit

currently:      speed 0.5       turn 1
```

u: for left circle/ i: forward / o: right circle / j: turn anti-clockwise / K: stop all motors / l: turn clockwise / m: left circle backwards / ,: straight backwards / .: right circle backward

2. Move the Higgs 2.0

1. To move the Higgs first all the steps from Point 3.7 from the Software section must be followed.
2. Develop the control code for the Higgs with the only requirement that It has to publish in the topic "cmd_vel" and the message is Twist type from geometry messages. Where the liner velocity is equal to msg.linear.x and the angular at msg.angular.z

```
ros::Subscriber<geometry_msgs::Twist> sub("cmd_vel", &messageCb );
speed_ang = msg.angular.z;
speed_lin = msg.linear.x;
```

3. Run the Ros Arduino serial package for the communication between the Nvidea Jetson and the Arduino.

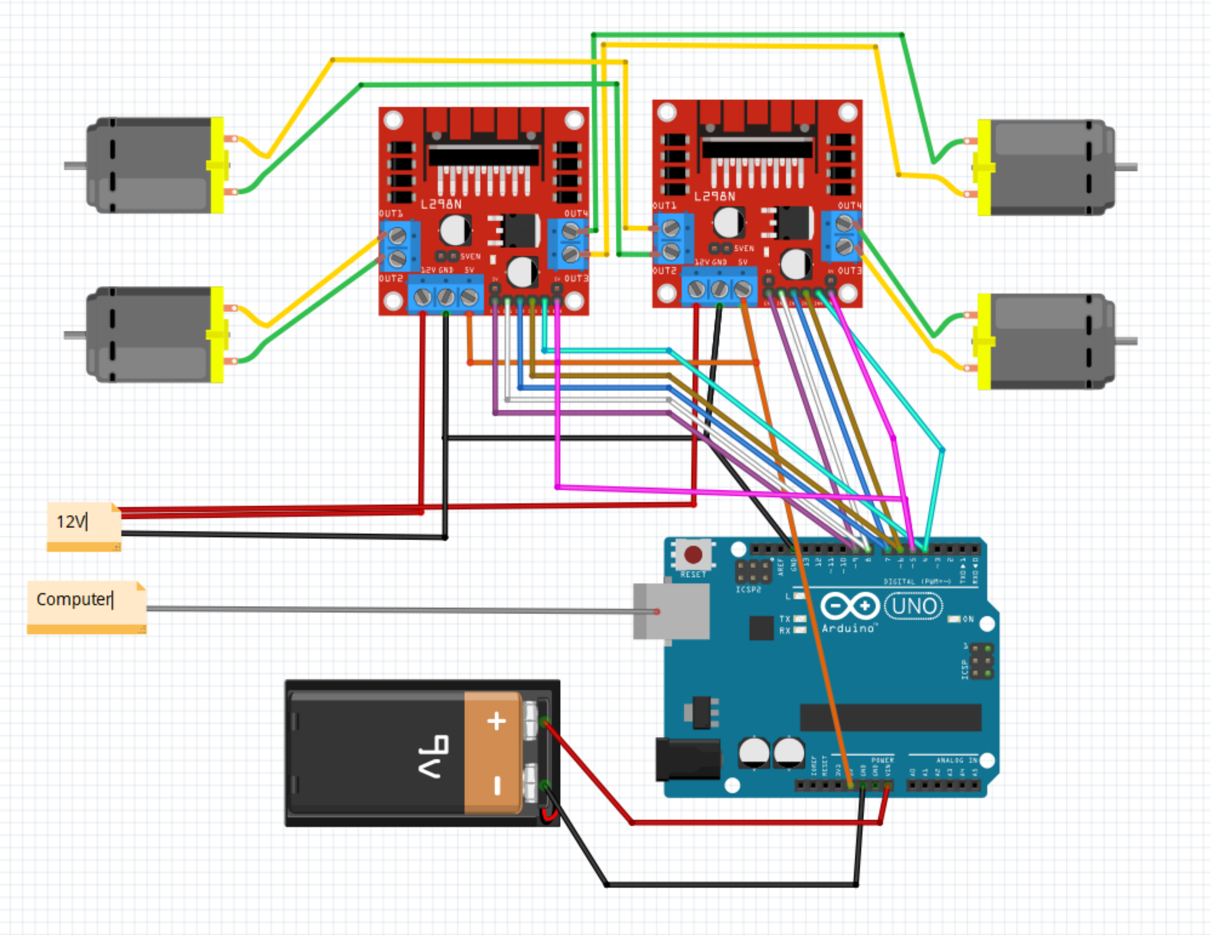
```
rosrun rosserial_python serial_node.py /dev/XXXXXXXX
```

Change the XXXXX with the port used by the Arduino.
You can find this information in the Arduino Ide **Tools/port** section.

4. Load the control and rosrun the control code.

3. Connections

The following picture describes how the current connection of the boards is made. Pay attention to the fact that the Arduino board can be powered using a cable connected to the computer or using an external battery, not by using both methods at the same time.



4. Maintenance

The maintenance of the robot is very simple, the user must check the following things.

For any maintenance reparation, firstly remove any energy source.

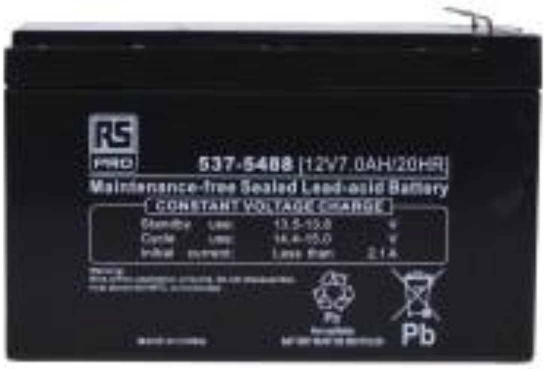
4.1. Fuse

The robot has installed a 20 Amp fuse, in case that the robot doesn't move, you first have to check the fuse. You can easily check the fuse with a visual inspection, but if a visual inspection is not enough you can check with a multi-meter if there is current flow between its clamps.

In case that the fuse is broke, the user must de-solder the current fuse, and weld the new one.

4.2. Batteries

The Higgs 2.0 has 3 batteries of 12 volts. In case that the robot doesn't move, after the visual inspection of the Fuse, is to completely charge the batteries for 6-8 hours. After this with a multi-meter check the voltage between positive and negative, it should be around 12 volts.



In case the batteries need to be replaced, you can find the same model at this LINK.

https://es.rs-online.com/web/p/baterias-de-plomo-acido/5375488?sra=pstk

4.3. Tyres

Before and after every run, a superficial inspection should be done, in case you notice that the tyre level is low , inflate them . If the problem persists , try to find a puncture and fix it.