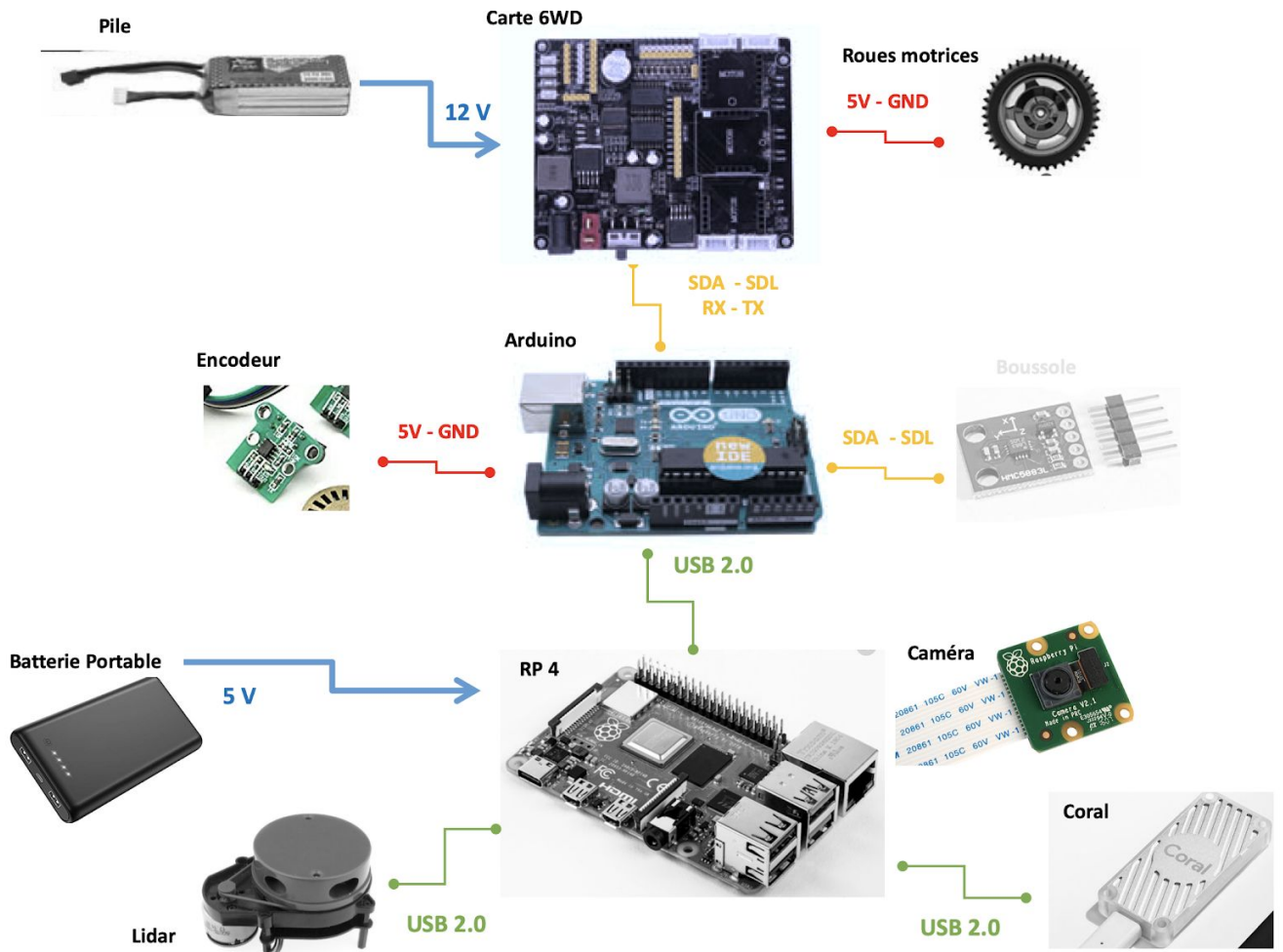


SLAMIT FROM SCRATCH :

Equipement du projet	2
Installation de l'os pour raspi 4	2
Recup Le dossier slamit du projet	3
Configuration de la RaspberryPi	3
Fichier où il faut changer l'adresse IP :	4
Configuration des Websockets - cmd Robot - 3min	4
Configuration de coral - 5 min	4
Installation ROS - 250 min	5
Configuration du lidar - 10 min	7
Lancer le projet	8
Arreter ROS	8

EQUIPEMENT DU PROJET



INSTALLATION DE L'OS POUR RASPI 4

- Télécharger [Raspbian Buster with desktop and recommended software](#)
- Utiliser BalenaEtcher pour configurer la carte SD

RECUP LE DOSSIER SLAMIT DU PROJET

```
git clone http://github.com/lmejbar/slamit.git
```

CONFIGURATION DE LA RASPBERRYPI

- Config mot de passe : raspberry
- Config Wifi
- Camera : enable
- SSH : enable
- VNC : enable
 - > Il faut aller dans les param de VNC (logo en haut à droite) et In the *Security* tab, set the *Encryption* to **Prefer off** and the *Authentication* to **VNC Password**.
 - > In the *Users & Permissions* tab, select the *Standard user (user)*, and click the **Password...** button. Set the password then click **OK**, then click **OK**.

- Installer Apache2 et php

```
sudo apt update
sudo apt upgrade
sudo apt install apache2
sudo chown -R pi:www-data /var/www/html/
sudo chmod -R 770 /var/www/html/
sudo apt install php -y
```

FICHER OÙ IL FAUT CHANGER L'ADRESSE IP :

- ~/slamit/rosbash (L9)
- /var/www/html/slamit.html (L18, 35, 118, 127)
- /var/www/html/connect.php (L7, 14)
- ~/slamit//Coral-video-browser/coral_webstreaming.py (L 160)

CONFIGURATION DES WEBSOCKETS - CMD ROBOT - 3MIN

```
cd ~/slamit/Websocket
rm exec

sudo apt install libwebsockets-dev
gcc -o exec webserial.c -lwebsockets
//Ajouter une ligne dans un fichier avec id arduino
sudo vim /etc/udev/rules.d/arduino.rules

    KERNEL=="ttyUSB*", ATTRS{idVendor}=="1a86",
    ATTRS{idProduct}=="7523" SYMLINK+="arduino"

sudo usermod -a -G dialout www-data
sudo udevadm trigger
sudo udev restart
```

CONFIGURATION DE CORAL - 5 MIN

```
echo "deb https://packages.cloud.google.com/apt coral-edgetpu-stable main" | sudo tee
/etc/apt/sources.list.d/coral-edgetpu.list

curl https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add -

sudo apt-get update

sudo apt-get install libedgetpu1-std

sudo apt-get install libedgetpu1-max

pip3 install imutils

pip3 install opencv-python==3.4.6.27

sudo apt install libatlas-base-dev

sudo apt install libjasper-dev
sudo apt install python3-edgetpu
```

INSTALLATION ROS - 250 MIN

Etape 1 : Installation installateur ROS - Environ 5 min

```
sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb_release -sc)
main" > /etc/apt/sources.list.d/ros-latest.list'

sudo apt-key adv --keyserver 'hkp://keyserver.ubuntu.com:80' --recv-key
C1CF6E31E6BADE8868B172B4F42ED6FBAB17C654

sudo apt-get update

sudo apt-get install -y python-rosdep python-rosinstall-generator python-wstool
python-rosinstall build-essential cmake
```

Etape 2 : Initialisation de l'outils ROS Dependencies - Environ 1 min

```
sudo rosdep init

rosdep update
```

Etape 3 : Création du répertoire des outils Environ 5 min

```
mkdir ~/ros_catkin_ws

cd ~/ros_catkin_ws

rosinstall_generator desktop --rosdistro melodic --deps --wet-only --tar >
melodic-desktop-wet.rosinstall

wstool init -j8 src melodic-desktop-wet.rosinstall

sudo apt-get install libogre-1.9-dev
```

Etape 4 : Installation des dépendances - Environ 60 min

```
cd ~/ros_catkin_ws

rosdep install --from-paths src --ignore-src --rosdistro melodic -y
```

Etape 5 : Installation des 186 outils de ROS - Environ 120 min

```
cd ~/ros_catkin_ws

sudo ./src/catkin/bin/catkin_make_isolated --install -DCMAKE_BUILD_TYPE=Release
--install-space /opt/ros/melodic -j2

source /opt/ros/melodic/setup.bash

echo "source /opt/ros/melodic/setup.bash" >> ~/.bashrc
```

Etape 6 : Création du répertoire Catkin - Environ 5 min

```
mkdir -p ~/catkin_ws/src

cd ~/catkin_ws/

sudo apt-get install python-catkin-tools

catkin_make

echo "source $HOME/catkin_ws/devel/setup.bash" >> ~/.bashrc
```

Etape 7 : Installation des packages additionnels - Environ 30 min

```
cd ~/catkin_ws/src

sudo git clone <link>

cd ..

sudo apt-get install python-rosdep

rosdep update

rosdep install --from-paths src --ignore-src -y

catkin_make
```

Lien Github qu'il faudra `git clone` :

Lidar	https://github.com/EAIBOT/ydlidar.git
Hector SLAM	https://github.com/tu-darmstadt-ros-pkg/hector_slam.git
Ros auth	https://github.com/GT-RAIL/rosauth.git
Rosbridge	https://github.com/RobotWebTools/rosbridge_suite.git
Rosserial	https://github.com/ros-drivers/roserial.git
Mjpeg server	https://github.com/RobotWebTools/mjpeg_server.git

CONFIGURATION DU LIDAR - 10 MIN

- Création de l'alias USB /deb/ydlidar

```
cd ~/catkin_ws/src
sudo git clone https://github.com/EAIBOT/ydlidar.git
cd ..
catkin_make
roscd ydlidar/startup
sudo chmod 777 ./*
sudo sh initenv.sh
sudo udevadm trigger
sudo chmod 777 /dev/ttyUSB0
sudo usermod -a -G www-data pi
sudo service apache2 restart
sudo usermod -a -G dialout pi
```

LANCER LE PROJET

```
cd slamit  
./<executable>
```

<exec>	Commande
Lancer ROS	./rosbash
Lancer AI Coral	./aibash
Lancer ROS & AI Coral	./rosaibash

ARRETER ROS

```
rosnode kill -a
```