The analog joystick of the gamepad returns at each movement of the gamepad x and y.

X=0, y=0 => motor cmd = 0 PWM = 0 and delta compass = 0 °

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Joystick data** | | | **Motion command for FRDM\_KV31 board** | | | |
| **X** | **Y** | **Motor cmd** | | **PWM** | **Delta compass** |
| 0 | 0 | 0 | | 0 | 0 ° |
| 0 | +32767 | 1 (front) | | 2499 | 0 ° |
| +32767 | +32767 | 1 (front) | | 2499 | +45 ° |
| +32767 | 0 | Previous motor cmd | | Previous PWM | Previous delta compass |
| +32767 | -32767 | 3 (back) | | 2499 | -45 ° |
| 0 | -32767 | 3 (back) | | 2499 | 0 ° |
| -32767 | -32767 | 3 (back) | | 2499 | +45 ° |
| -32767 | 0 | Previous motor cmd | | Previous PWM | Previous delta compass |
| -32767 | +32767 | 1 (front) | | 2499 | -45 ° |

# Hikey

## partition table

<http://snapshots.linaro.org/96boards/hikey/linaro/debian/18/>

# Partition table of Internal flash

see https://github.com/96boards/l-loader

* - vrl
* - vrl\_backup
* - mcuimage
* - fastboot
* - nvme
* - boot (grub+UEFI)
* - reserved (not used if debian)
* - cache (not used if debian)
* - system (rootfs of debian withy kernel)

# Create Bootable SD Card

This section show how to install the Debian operating system that is bootable from the SD Card to your HiKey on a Linux host computer.

## Step 1: board recovery

apply the procedure of 96 board

see <https://www.96boards.org/documentation/consumer/hikey/installation/board-recovery.md.html>

## Step 2: Download SD Card Image

* Download

wget <http://snapshots.linaro.org/96boards/hikey/linaro/debian/latest/linaro-stretch-developer-hikey->\*.sd.gz -O linaro-stretch-developer-hikey.sd.gz

* Decompress

$ gunzip linaro-stretch-developer-hikey.sd.gz

## Step 3: Flash image to SD Card

* Remove SD card from host computer and run the following command:

$ lsblk

* Note all recognized disk names
* **Insert SD card** and run the following command (again):

$ lsblk

* Note the newly recognized disk. This will be your SD card.
* **Remember** your SD card device name
* Flash SD card

$ dd if=[linaro-stretch-developer-hikey.sd](http://linaro-stretch-developer-hikey.sd/) of=/dev/xxx status=progress

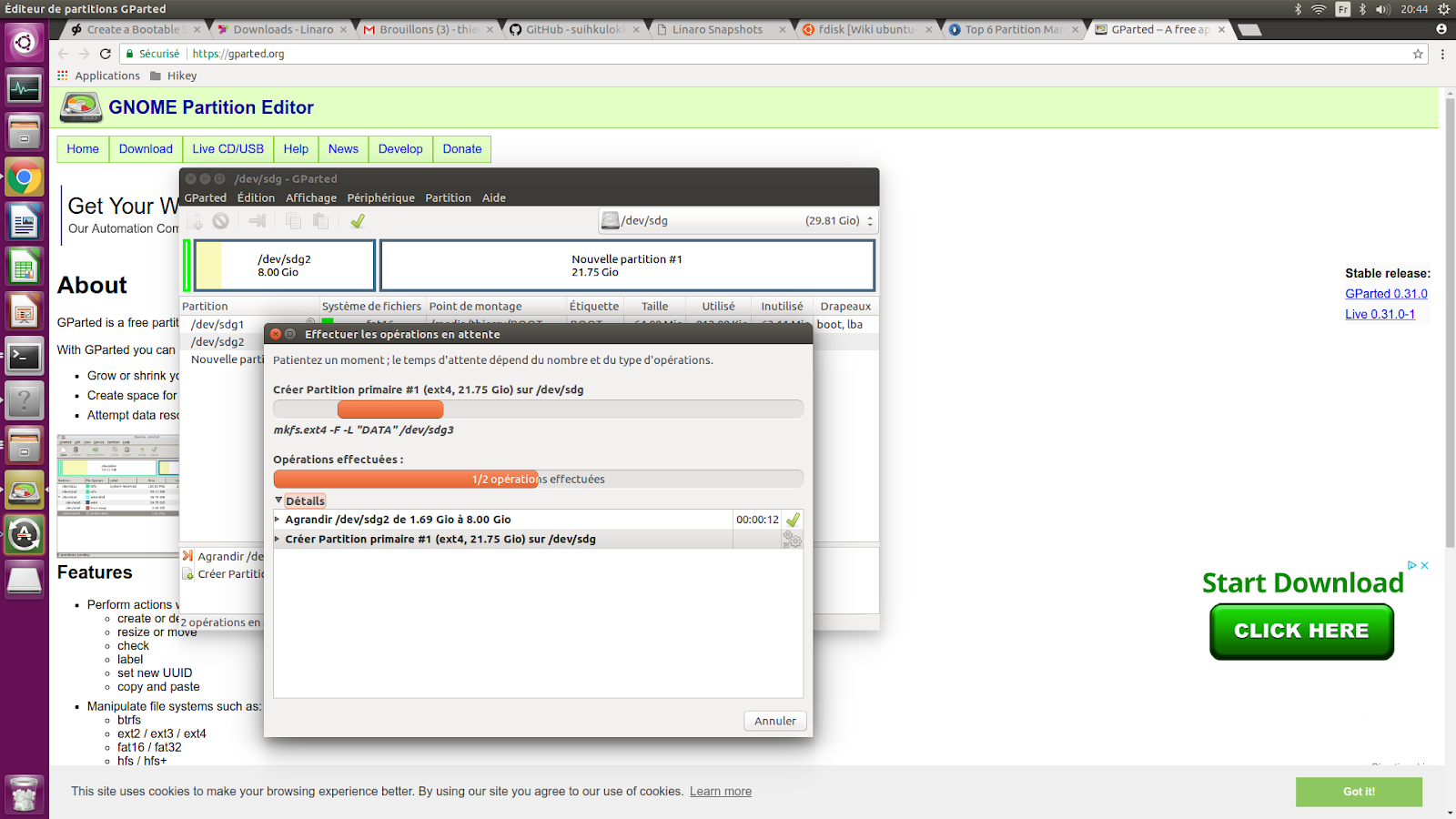
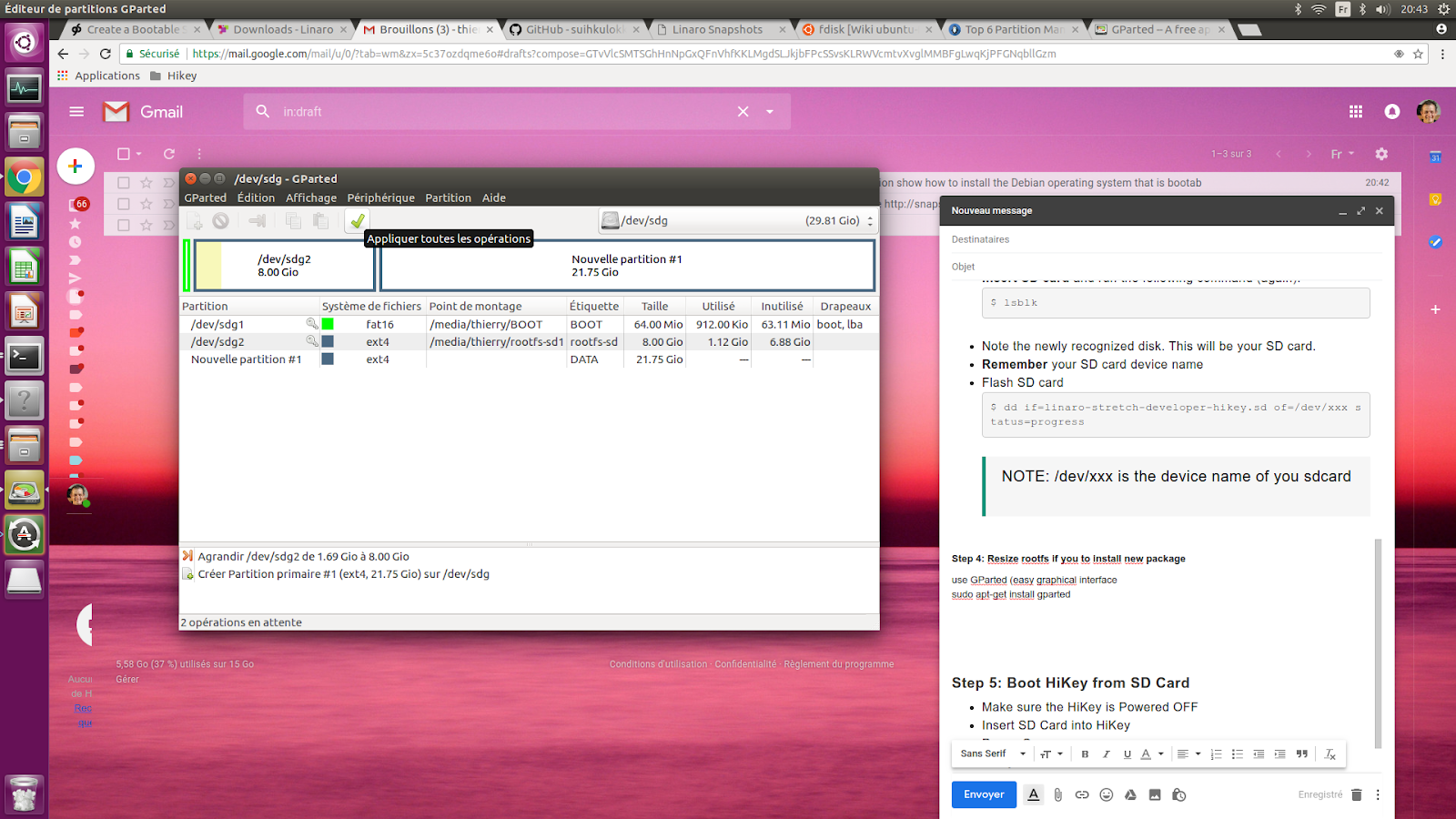
NOTE: /dev/xxx is the device name of you sdcard

## Step 4: Resize rootfs if you to have enough space in rootfs to install new package

* use GParted (easy graphical interface, installation on you workstation sudo apt-get install gparted)
* fix the size to 8 Go (8192 Mo) or more, it depends on the SD card memory size
* use the same tool to create ext4 data partition named "DATA" at the end of SD card

now the SD card is ready and this is the list of SD partitions:

* - boot (grub+UEFI)
* - rootf (debian stretch - linux 4.15 )
* - DATA



## Step 5: Boot HiKey from SD Card

* Make sure the HiKey is Powered OFF
* Insert SD Card into HiKey
* Plug HDMI
* Plug keyboard on one USB
* Open a terminal and monitor activity on the serial link (use tool minicom)
* Power On
* Hikey should now boot Debian from SD Card

## Step 6: configure keymap (azerty) (by default "us" => querty)

$ setxkbmap fr

## Step 7 Wireless Network configuration

The HiKey board includes built in [2.4GHz IEEE802.11 b/g/n WiFi networking](http://www.ti.com/product/WL1835MOD). The board does not support the 5GHz band. To use the wireless LAN interface for the first time (or to switch wireless networks) you should click on the wireless LAN icon on the bottom right of the desktop display. The yellow LED between the microUSB and the Type A USB on the front board edge indicates wireless network activity.

You can configure the network from UI, or manually from console:

$ sudo nmtui

Select ‘Activate a connection’, Choose your WiFi access point (SSID) and fill the relevant information (password, etc…)

You can check network status by using this command.

$ sudo nmcli dev wifi

## Step 8: update rootfs

$ sudo apt-get update

$ sudo apt-get upgrade

$ sudo apt-get dist-upgrade

there is no problem to update the distribution (2018/06/25)

## Step 9: Install KDE for stretch

see https://wiki.debian.org/fr/DebianKDE

$ sudo apt-get install apt aptitude tasksel

$ sudo aptitude install ~t^desktop$ ~t^kde-desktop$

# Save SD Card

$ sudo cat /dev/xxx > save\_image.img

# gestion disque partition permet de voir les partion de la flash emmc de la hikey board

// permet de faire de disque image de la flash

// https://packages.debian.org/fr/jessie/gnome-disk-utility

sudo apt-get install gnome-disk-utility

# Install eclipse

List of package required for the project

* eclipse (java)
* eclipse-cdt (C and C++)
* eclipse-cdt (C and C++ with autotools)
* eclipse-anyedit (if wyou want to use eclipse to edit any type of files)
* eclipse-egit (git)

sudo apt-get install -y eclipse

sudo apt-get install -y eclipse-cdt

sudo apt-get install -y eclipse-cdt-autotools

sudo apt-get install -y eclipse-anyedit

sudo apt-get install -y eclipse-egit

# List hardwardsup

sudo apt-get install lshw

# OLD info

# bluethooth pour LX music

sudo apt-get install xmms2 xmms2-plugin-\*

//bluethooth cmd:List of devices to get the MAC address of my device:bt-device -l

sudo apt-get install bluez-tools

- open cv (open soucre vision stack)

https://github.com/milq/milq/blob/master/scripts/bash/install-opencv.sh

sudo apt-get -y update

#sudo apt-get -y upgrade

#sudo apt-get -y dist-upgrade

#sudo apt-get -y autoremove

# 2. INSTALL THE DEPENDENCIES

# Build tools:

sudo apt-get install -y build-essential cmake

# GUI (if you want to use GTK instead of Qt, replace 'qt5-default' with 'libgtkglext1-dev' and remove '-DWITH\_QT=ON' option in CMake):

sudo apt-get install -y qt5-default libvtk6-dev

# Media I/O:

sudo apt-get install -y zlib1g-dev libjpeg-dev libwebp-dev libpng-dev libtiff5-dev libjasper-dev libopenexr-dev libgdal-dev

# Video I/O:

sudo apt-get install -y libdc1394-22-dev libavcodec-dev libavformat-dev libswscale-dev libtheora-dev libvorbis-dev libxvidcore-dev libx264-dev yasm libopencore-amrnb-dev libopencore-amrwb-dev libv4l-dev libxine2-dev

# Parallelism and linear algebra libraries:

sudo apt-get install -y libtbb-dev libeigen3-dev

# Python:

sudo apt-get install -y python-dev python-tk python-numpy python3-dev python3-tk python3-numpy

# Java:

sudo apt-get install -y ant default-jdk

# Documentation:

sudo apt-get install -y doxygen

#++ T.GAUTIER

sudo apt-get install FFmpeg

# 3. INSTALL THE LIBRARY (YOU CAN CHANGE '3.2.0' FOR THE LAST STABLE VERSION)

sudo apt-get install -y unzip wget

wget https://github.com/opencv/opencv/archive/3.3.0.zip

unzip 3.3.0.zip

wget -O opencv\_contrib.zip https://github.com/Itseez/opencv\_contrib/archive/3.3.0.zip

unzip opencv\_contrib.zip

cd opencv-3.3.0

mkdir build

cd build

cmake -Wno-dev -D CMAKE\_BUILD\_TYPE=RELEASE -DCMAKE\_INSTALL\_PREFIX=/usr/local -DOPENCV\_EXTRA\_MODULES\_PATH=../../opencv\_contrib-3.3.0/modules ../ >cmake\_result.txt

make -j4

sudo make install

sudo ldconfig

# 4. EXECUTE SOME OPENCV EXAMPLES AND COMPILE A DEMONSTRATION

# project generated in C++ with eclipse add path of open cv librairy

https://docs.opencv.org/trunk/d7/d16/tutorial\_linux\_eclipse.html

export LD\_LIBRARY\_PATH=$LD\_LIBRARY\_PATH:/usr/local/lib

//installer package hwpack

//voir https://doc.ubuntu-fr.org/dpkg

//se placer dans le répertoire ou il ya tout les fichiers \*.deb

sudo dpkg -i -R ./

//dpkg ne gère pas les dépendances, il faut ensuite, pour compléter l'installation d'une application possédant des dépendances, exécuter dans un terminal la commande:

sudo apt-get -f install

//sound http://www.alsa-project.org/alsa-doc/alsa-lib/examples.html

sudo apt-get install libasound2

sudo apt-get install libasound2-dev

sudo apt-get install libasound2-doc

//sudo apt-get install Alsa-base

- synthèse vocale

//https://www.gnu.org/software/gnuspeech/

//https://tuxicoman.jesuislibre.net/2015/05/synthese-vocale-sous-linux.html

//http://espeak.sourceforge.net/mbrola.html not free

//http://www.pobot.org/Synthese-vocale-avec-espeak-et.html?lang=fr

//sudo apt-get install espeak mbrola mbrola-fr4

//mbrola et mbrola-fr4 non présent sur ARM debian voir next URL

//http://www.tcts.fms.ac.be/synthesis/mbrola/mbrcopybin.html

// festival (1:2.1~release-8)

//gstreamer0.10-pocketsphinx (0.8-5) Speech recognition tool - gstreamer plugin

//pocketsphinx (0.8-5) Speech recognition tool

//install de espeak

sudo apt-get install espeak

sudo apt-get install espeak-data

sudo apt-get install festival

//install de mbrola

wget http://tcts.fpms.ac.be/synthesis/mbrola/bin/armlinux/mbrola.rar

sudo apt-get install unrar

unrar e mbrola.rar

sudo mv mbrola /usr/bin

sudo chmod 777 /usr/bin/mbrola

//problem executuion de mbrola => test conversion dos vers uinx pas de changement!!!!

sudo apt-get install dos2unix

linaro@linaro-alip:/usr/bin$ sudo dos2unix mbrola

//install de langue française

sudo mkdir /usr/share/mbrola

sudo mkdir /usr/share/mbrola/voices

wget http://tcts.fpms.ac.be/synthesis/mbrola/dba/fr1/fr1-990204.zip

unzip fr1\*.zip

sudo mv fr1/fr1 /usr/share/mbrola/voices/

//test

# echo "salut les amis, c'est Pobot" > exemple.txt

# espeak -v mb/mb-fr1 -f exemple.txt | mbrola /usr/share/mbrola/voices/fr1 - - | aplay -r16000 -fS16

# Annexe 1: link

link to understand the different name of rootfs for 96 boards

https://wiki.linaro.org/Platform/DevPlatform/Rootfs

<https://releases.linaro.org/debian/images/alip-arm64/17.02/>

http://snapshots.linaro.org/debian/images/stretch/alip-arm64/89/

to test new graphic env

https://releases.linaro.org/debian/images/developer-arm64/18.04/

<https://snapshots.linaro.org/96boards/hikey/linaro/debian/18/>

rootfs-linaro-stretch-developer-hikey-20180521-18.img.gz

* video problem must be more test
* apt-get upgrade and apt-get distupgrade wok

1. reference platform roofs:

http://snapshots.linaro.org/reference-platform/embedded/morty/hikey/latest/rpb/

<https://releases.linaro.org/reference-platform/embedded/hikey/16.12>

* rpb-console-image-hikey-20170207171723-29.rootfs.ext4
* rpb-desktop-image-hikey-20170207172341-29.rootfs.ext4
* rpb-weston-image-hikey-20170207171723-29.rootfs.ext4

cat /etc/os-release

give the identification of the os

cat /etc/build

list all meta used to generate image with poky (yocto)

**test 1 rpb-weston-image-hikey**

sudo fastboot flash system rpb-weston-image-hikey-20171028131308-85.rootfs.img

video fonctionne correctement bureau minimale avec la commande

weston

pas de commade apt-get , dpkg c’est une distribution générée sous yocto au vue du fichier /etc/os-release

opkg

**test 2 rpb-destop-image-hikey**

package video

- sudo apt-get install xwayland

- sudo apt-get install weston

List of directories of rootfs debian of hikey board

* /bin
* /boot
* /dev (empty)
* /etc
* /home
* /lib
* /media (empty)
* /mnt (empty)
* /proc (empty)
* /run (empty)
* /sbin
* /sys (empty)
* /tmp (empty)
* /usr
* /var

/var/lib/opkg/info directory

list of all packges install, all files <name>.control (like 96boards-tools.control) give the descrition of a package install:

* Package: name of the package
* Version: version of the package
* Description: details information of the package
* Section:
* Priority:
* Maintainer: organization in charge to develop and maintain this software project
* License: reference to the license used for this software project
* Architecture: architecture supported
* OE:
* Homepage: url where finding all info of this package
* Depends: list of lib required with the revision limit
* Source: url where finding the source of this software project