

Créer sa plate-forme Astro-photo avec un Raspberry 4



Robert M. - Mars 2023

Table des matières

1- Matériel requis.....	3
2- Création du système.....	3
2.1 Installation de l'OS Ubuntu-Mate et des logiciels.....	4
2.2 Premier boot.....	5
2.2.1 Mise à jour de l'heure système.....	6
2.3 Installation de Kstars-Ekos, Indi et gsc.....	6
2.4 Configuration de Kstars.....	6
2.4.1 Configuration de Indi et Ekos.....	7
2.4.2 Installer les index d'astrométrie.....	7
2.4.3 Définition d'un profil dans Kstars.....	8
2.5 Installer PHD2.....	8
3- Mise en œuvre de la plateforme.....	8
3.1 Connexion du matériel.....	8
3.2 Connexion entre le RPI4 et le PC client.....	8
3.2.1 Connexion par RJ45.....	8
3.2.2 Connexion par hotspot.....	9
3.2.3 Améliorer la portée du WiFi du RPI.....	10
4- Boîtier d'alimentation DIY.....	11
5- Une autre solution.....	11

Avec l'apparition des Raspberry, l'astronome amateur a rapidement développé des solutions de pilotage de sa configuration à distance. Ce micro-ordinateur, s'est rapidement imposé grâce à son coût modique, sa facilité de mise en œuvre. Des solutions logicielles freewares et commerciales sont apparues:

- NAFABOX, développé par un français, freeware
<https://share.obspm.fr/index.php/s/WowXMAZkHPd3K9j>
- ASTROBERRY, <https://www.astroberry.io/>, freeware
- ASIAIR© développé par ZWO, <https://astronomy-imaging-camera.com/asiair>, commercial,
STELLARMATE© par les auteurs de KSTARS-EKOS, <https://www.stellarmate.com/>, commercial,
- FAIT MAISON.

Toutes ces solutions s'appuient sur l'OS Linux et la solution d'astrophotographie KSTARS-EKOS, PHD2 et autres logiciels.

A cette date (mars 2023), la solution ASTROBERRY est dans une impasse. Conçu avec un OS 32 bits, KSTARS ne peut plus être mis à jour, sa version 32 bits n'étant plus supportée depuis la 3.6.0.

NAFABOX semble être en sommeil, bien que toujours fonctionnel.

ASIAIR© est la solution de ZWO qui inclut un boîtier RPI4 avec 4 sorties 12V et une antenne wifi externe (250€ ou 380€ selon la version).

STELLARMATE© est une solution logiciel commerciale pour RPI4 (49\$) ou matériel avec Stellarmate X et Stellarmate Pro. La première est basée sur un mini-PC (299\$ ou 329\$ selon la mémoire), la seconde sur un RPI4 plus 4 sorties 12V, 2 sorties 12V pour résistance chauffante, un GPS intégré, un contrôleur pas à pas (499\$).

Et si vous êtes curieux et bricoleur, vous pouvez monter par vous-même une plate-forme. C'est cette dernière solution qui est décrite dans ce document.

1- Matériel requis

- 1 RPI4 2 Go minimum, dans un boîtier avec un ventilateur, une alimentation.
- 1 carte SD 32 Go minimum, 64 Go ou plus recommandé ou 1 clé USB 32 Go minimum, 64 Go ou plus recommandé,
- 1 clavier, 1 souris, un écran avec prise HDMI ou DVI avec convertisseur HDMI-VDI, pour configurer le RPI4.

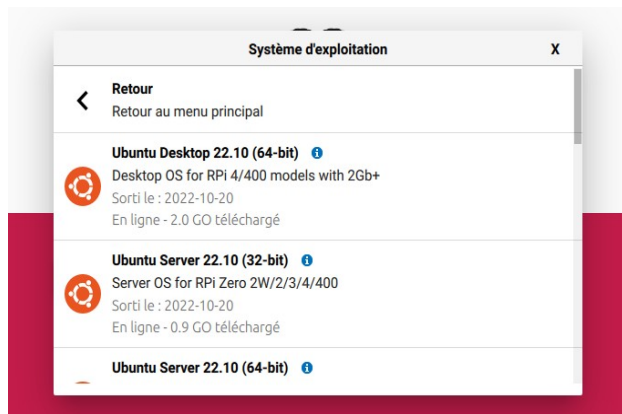
2- Création du système

Il s'agit de créer une carte SD/Clé USB avec un OS et les logiciels d'astrophotos nécessaires pour une séance de capture en ciel profond. Pour cela nous utiliserons :

- Ubuntu-Mate comme OS (Linux)

- Kstars-Ekos, Indi, PHD2, Astap, libre à vous d'allonger cette liste.

2.1 Installation de l'OS Ubuntu-Mate et des logiciels



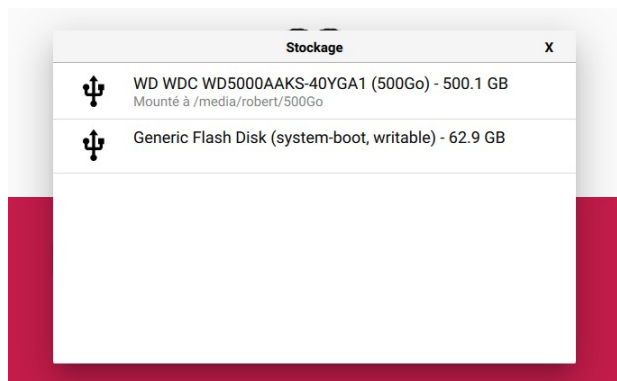
Commencer par télécharger sur un PC pour votre OS, l'utilitaire *Pi Imager* ici : <https://www.raspberrypi.com/software/>

Après installation, lancez l'utilitaire.

Cliquez sur *CHOISISSEZ L'OS*. Puis sur *Other general purpose OS*, puis sur *Ubuntu*, puis sur *Ubuntu Desktop 22.10 (64bits)*.

Après avoir insérer une carte SD, ou une clé USB, sur votre PC, cliquez sur *CHOISISSEZ LE*

STOCKAGE.



Dans l'exemple, Generic Flash Disk (system-boot,writable) – 62,9 GB.

Puis sur *ECRIRE*. L'OS va alors être gravé sur la carte. Attendez la fin du processus.

2.2 Premier boot

Insérez votre carte SD fraîchement gravée, ou votre clé USB (sans la carte SD), dans le RPI4.

Branchez votre RPI sur son alimentation (5V, 3A), connectez un clavier USB, une souris USB et un écran sur le port mini-HDMI avec le câble fourni. Si vous ne disposez pas d'un port HDMI sur un écran, un convertisseur HDMI-VDI, permet de connecter le RPI à l'écran. Reliez aussi le RPI à votre réseau par un câble Ethernet.

Une première configuration vous est proposée : choix du pays, du langage, du clavier, configuration du wifi, ect. Le boot se termine et l'interface graphique de Ubuntu-Mate apparaît. Vous ne serez pas dépaycé, avec une barre de menu en haut de l'écran, quelques icônes sur le bureau et le curseur de souris.



L'icône de gauche déroule un menu arborescent organisé par fonctions : Programmation, Education, Internet, etc.

Pour placer un programme dans la barre de menu ou sur le bureau, déroulez le menu, allez sur le programme, faites un clic droit et choisissez *Bureau* ou *Tableau de Bord*. Installez le navigateur Firefox, le gestionnaire de fichiers Caja, le terminal Mate au minimum.

Dans le menu *Préférences*, cliquez sur *Mate Tweak*. Dans la fenêtre qui s'ouvre, cliquez sur *Tableau de bord*. Dans la section *Tableaux de Bord*, choisissez une des configurations. *Traditionnel* est mon choix, tel qu'il apparaît sur l'image ci-dessus.

Ouvrez un terminal (par CTRL-ALT-T) et installez l'éditeur de texte graphique Pluma :

```
sudo apt install pluma
```


La commande *sudo* (superviseur do) demande le mot de passe administrateur configuré lors du premier boot.

2.2.1 Mise à jour de l'heure système.

Le RPI4 ne dispose pas d'une horloge interne. Pour avoir une heure et date système à jour, plusieurs méthodes sont possibles :

- Manuelle: Ouvrez un terminal (par Ctrl-Alt-T) et tapez la commande suivante :
- ***sudo date -s "aaaa-mm-jjThh:mm:ss"*** et validez par Entrée.
- Par GPS: Si vous utilisez un GPS UBS, il mettra à jour dans Kstars, l'heure/date et la localisation (les coordonnées), mais pas l'heure et la date système. Voir l'appendice pour la mise à jour de l'heure/date système avec le GPS.
- Matériel: On peut se procurer un système d'horloge dit RTC et l'installer à demeure dans le RPI4.
- Avec un smartphone équipé de Stellarmate. Une fois connecté par la liaison wifi ou RJ45, l'heure, de la date et des coordonnées du lieu sont mises à jour dans Kstars et appliqués à la monture.

2.3 Installation de Kstars-Ekos, Indi et gsc.

Pour cela nous allons suivre la procédure décrite sur le site <http://indilib.org>, menu Get Indi – Ubuntu. Ouvrez un terminal en cliquant sur l'icône  ou par Ctrl-Alt-T.

Une fenêtre s'ouvre, analogue à une fenêtre MS-DOS de Windows. Tapez successivement les 2 lignes de commande suivantes et terminez par un appui sur la touche Entrée à chaque fois.

- 1) *sudo apt-add-repository ppa:mutlaqja/ppa*
- 2) *sudo apt update && sudo apt install indi-full gsc kstars-bleeding*

La première commande référence le dépôt mutlaqja/ppa comme une nouvelle source de logiciels. Lors des mise à jour, il sera pris en compte et Kstars sera mis à jour.

La deuxième commande va installer successivement le serveur Indi et tous ses pilotes matériels, gsc un ciel virtuel qui permettra de simuler toute une session d'astrophoto dans Kstars, et enfin Kstars-Ekos lui-même.

Une fois l'installation terminée, vous trouverez Kstars-Ekos dans le menu *Education*. En cliquant droit sur l'entrée du Menu, vous pourrez installer une icône de lancement de Kstars sur le bureau ou dans la barre de menu.

2.4 Configuration de Kstars

Lancez Kstars. Ouvrez le menu *Configuration – Configure Kstars*. Je ne décrirais que ce qui est propre à Indi et Ekos. Pour le reste, reportez vous à la documentation française de Kstars. La dernière version est disponible à cette adresse : <https://www.webastro.net/forums/topic/238633-manuel-de-kstars-v363/#comment-3062916>.

2.4.1 Configuration de Indi et Ekos

Cliquez sur l'icône *Indi*. Laissez les répertoires par défaut tel quel. Dans la section *Times & Location*, choisissez la méthode de mise à jour de l'heure et la localisation de Kstars :

- Par Kstars, l'heure et la date sont configurable dans la barre de menu.
- Par la monture, ce sont les données de la monture qui seront rapatriées dans Kstars.
- Par GPS. Il faut utiliser une clé USB GPS, le pilote Indi GPSD. Installer gpsd et gpsd-clients, puis configurer le fichier:

sudo apt install gpsd gpsd-clients

Dans un terminal, tapez ***sudo pluma /etc/default/gpsd*** et modifiez le fichier comme suit:

```
# Default settings for the gpsd init script and the hotplug wrapper.
# Start the gpsd daemon automatically at boot time
START_DAEMON="true"
# Use USB hotplugging to add new USB devices automatically to the daemon
USB AUTO="true"
# Devices gpsd should collect to at boot time.
# They need to be read/writeable, either by user gpsd or the group dialout.
DEVICES="/dev/ttyACM0"
# Other options you want to pass to gpsd
GPSD_OPTIONS="-n"
```

Cliquez la case *Fenêtre indépendante*.

Cliquez sur l'icône *Ekos*. Cliquez sur la case *A gauche* et *Fenêtre indépendante*. Par défaut la barre d'icône d'Ekos est horizontale en haut de la fenêtre Ekos. On peut aussi la positionner verticalement à gauche de la fenêtre d'Ekos. A l'usage c'est la disposition la plus pratique.

2.4.2 Installer les index d'astrométrie.

Kstars-Ekos possède des fonctions d'astrométries, pour lesquels il est nécessaire de disposer de fichiers d'index. 4 systèmes sont proposés à l'utilisateur :

- Un système interne à Kstars, avec les fichiers d'index d'Astrometry.net,
- Astrometry.net et ses fichiers d'index. Sur une clé USB 64 Go, on peut installer les 30 Go de fichiers d'index pour tous les champs de vision.
- Astap avec son propre fichier d'index. Astap n'est pas installé par défaut. Pour l'installer il faut aller sur le site d'Astap <https://www.hnsky.org/astap.htm>. Vous téléchargerez l'exécutable et le fichier d'index qui vous convient, puis installez les sur le RPI4.
- Watney avec son propre fichier d'index. Pour l'installer il faut aller sur le site de Watney: <https://github.com/Jusas/WatneyAstrometry/releases> . Dans la rubrique Assets, choisissez la version CLI pour RPI4, arm64. Par ailleurs vous devrez télécharger un fichier d'index: <https://github.com/Jusas/WatneyAstrometry/releases/tag/watneyqdb3>
Décompressez les deux fichiers dans un répertoire. Dans les options de l'onglet Alignement, vous indiquerez le répertoire où vous avez placé Watney.

Pour télécharger les fichiers d'index d'Astrometry.net, on peut le faire partir de Kstars lui-même, dans les options de l'onglet *Alignement*. Reportez-vous à la documentation française de Kstars.

2.4.3 Définition d'un profil dans Kstars

Pour pouvoir effectuer une séance d'astrophoto, il vous faut définir avec quels matériels vous allez travailler. Ce document n'ayant pas pour but de décrire le fonctionnement détaillé de Kstars, veuillez vous reporter à la documentation française de Kstars.

Une fois toutes ces étapes terminées, vous pourrez rajouter les logiciels d'astrophoto qui vous importent comme par exemple PHD2.

2.5 Installer PHD2

Exécutez les instructions ci-après dans un terminal:

```
sudo add-apt-repository ppa:pch/phd2
sudo apt update
sudo apt install phd2
```

PHD2 est placé dans le menu *Education*. Par un clic droit, on peut rajouter le programme dans la barre de menu et/ou sur le bureau.

3- Mise en œuvre de la plateforme

3.1 Connexion du matériel.

Le RPI4 dispose de 2 port USB3 et 2 port USB2. Les caméras seront de préférence connectées sur les ports USB3. Généralement, on a au maximum 2 caméras : une pour le guidage, l'autre pour imager. Selon le nombre de matériel utilisé, monture, roue à filtre, moteur de mise au point, GPS, Clé USB ou disque SSD de démarrage, etc, on pourra rajouter un petit hub USB avec 4 ports supplémentaires, alimenté de préférence, selon les matériels connectés. Comme les 2 ports USB3 sont déjà occupés par les caméras, utiliser un hub USB2 uniquement. Un HUB USB3 sur un USB2 crée des problèmes de connexion.

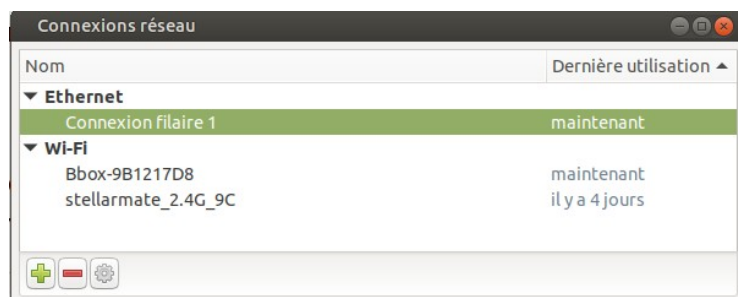
3.2 Connexion entre le RPI4 et le PC client

Lorsqu'on parle de PC client, ce peut être aussi une tablette ou un smartphone. Le PC client peut être sous Linux, iOS ou Windows.

Il existe de multiples façons de se connecter au RPI pour le piloter. Nous allons seulement évoquer le cas d'une séance en rase campagne, sans Internet disponible. Mais c'est utilisable en tout lieu, chez vous en particulier.

- Par un câble Ethernet RJ45
- Par un hotspot WiFi.

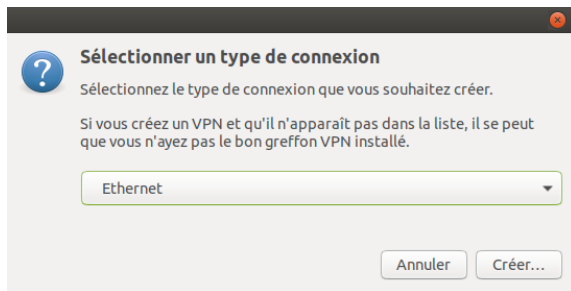
3.2.1 Connexion par RJ45



Un câble RJ45, de classe 5 minimum, et 100 mètres de longueur maximum, relie le RPI4 au PC client. D'un point de vue

informatique, on utilisera une liaison point-à-point sur le RPI et le PC. Très facile à configurer sous Linux. Je n'ai pas de système Windows, ni de système Mac. Consultez Internet pour ces deux cas.

Sur le RPI4/PC linux client, ouvrir le gestionnaire de réseau .

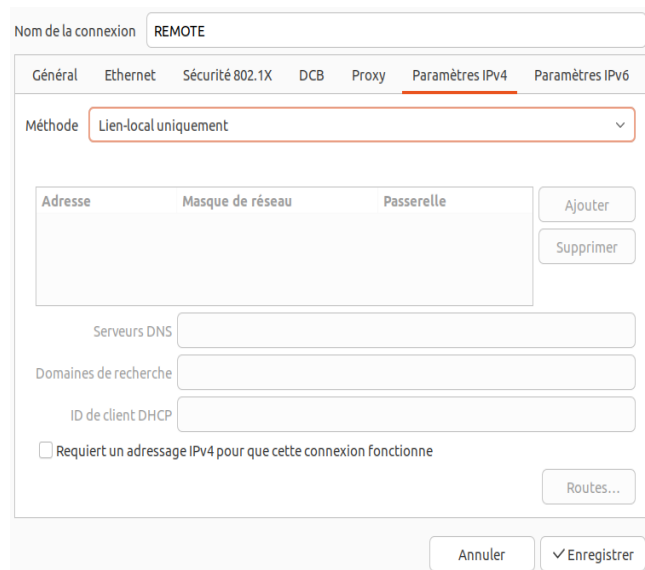
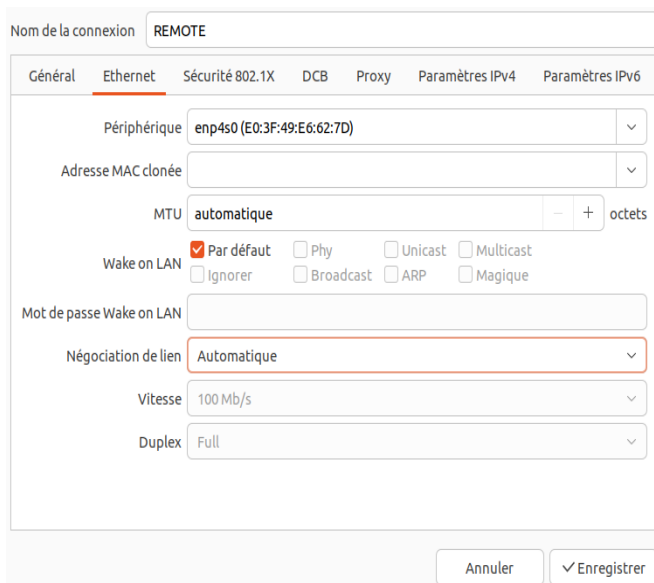


Cliquez sur + pour créer une nouvelle connexion. .

Choisissez **Ethernet**.

Dans l'onglet général donnez un nom à votre liaison, comme REMOTE par exemple.

Choisissez le périphérique Ethernet. Dans Link négociation, choisir **Automatique**.



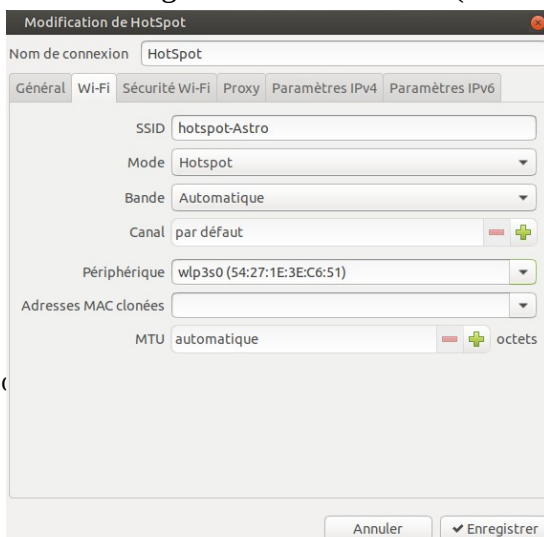
Dans l'onglet Paramètre IPV4, choisir **Lien-local uniquement**. Validez.

Une fois le RPI démarré, il va automatiquement se connecter avec la liaison REMOTE. Du côté PC client, choisissez la connexion REMOTE. Vous êtes prêt à prendre le contrôle du RPI.

Remarque: Lorsqu'on se connecte par câble à un réseau Internet, par une box par exemple, il faut aller dans le paramétrage de la liaison créée et configurer le bon périphérique réseau, Eth0 en général.

3.2.2 Connexion par hotspot

Il s'agit de créer un hotspot sur le RPI et de s'y connecter avec le PC client. Pour cela il suffit d'ouvrir le gestionnaire de réseau (menu *Système-Préférences-Internet & Réseaux- Configuration réseau avancée* ou clic-gauche sur l'icône réseau en haut à droite de l'écran puis *Modifier les connexions*) sur le RPI4 et de créer une nouvelle liaison wifi:



Renseigner un nom de connexion.

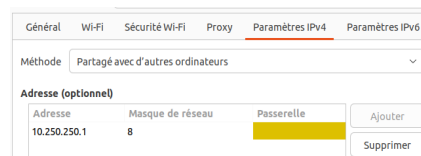
Renseigner un identifiant SSID.

Choisir le mode **Hotspot**.

Choisir le périphérique sans fil, ici wlp3s0, le plus souvent ce sera wlan0.

Dans l'onglet IPV4, ajouter une adresse comme 10.250.250.1.

Enregistrez et rebootez. C'est tout.

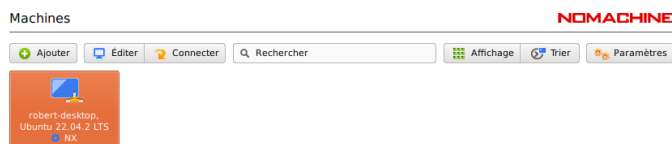


Pour relier le RPI au PC client, vous allez avoir besoin d'un logiciel de contrôle à distance comme VNC ou Nomachine. Aucun des deux n'est un logiciel libre, même s'il est gratuit. Sur le poste client vous devrez installer le client VNC/Nomachine selon votre OS. En effet le poste client peut utiliser n'importe quel OS du moment qu'il dispose d'un client VNC/Nomachine. Sur le RPI, il vous faut installer un serveur VNC, comme tightvnc ou x11vnc. Nomachine est plus simple à installer et d'utilisation. Je ne décris que l'installation de Nomachine. Pour VNC, veuillez vous reporter à la documentation de votre OS.

Nomachine doit être installé à partir de son site Web sur le RPI et le poste client : <https://downloads.nomachine.com/fr/download/?id=105&distro=Raspberry&hw=Pi4> . Choisir la version *nomachine_8.4.2_1_arm64.deb*. Après son installation, il est accessible dans la barre de menu à droite. Installer Nomachine sur le PC client/tablette/smartphone pour son OS.

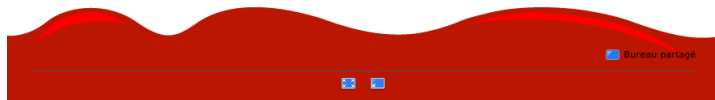
Un des avantages de Nomachine et de pouvoir copier un fichier du serveur vers le client ou l'inverse. Nomachine détecte automatiquement les connexions WiFi. Dès que vous êtes connecté au hotspot du RPI, à l'ouverture de Nomachine sur le PC client, la liaison apparaît sur le bureau de l'application.

Connectez-vous au wifi HotSpot. Cliquez sur *Affichez la fenêtre principale*, puis validez les 2 fenêtres qui apparaissent. La fenêtre intitulé *Machines*, affiche les matériels disponibles pour une connexion



Double cliquez sur l'icône du matériel détecté.

Vous pouvez maintenant piloter le RPI à distance avec votre PC client, ou une tablette ou un smartphone équipé du client Nomachine.



3.2.3 Améliorer la portée du WiFi du RPI.

Le WiFi du RPI4 n'est pas très puissant, sa portée est limitée. L'utilisation d'un répéteur WiFi comme un GL-iNet© améliore grandement la portée du WiFi. J'en utilise un, alimenté en 5V avec

une batterie de téléphone. Placer le répéteur sur l'araignée de la monture, afin que son orientation reste constante.

4- Boîtier d'alimentation DIY.

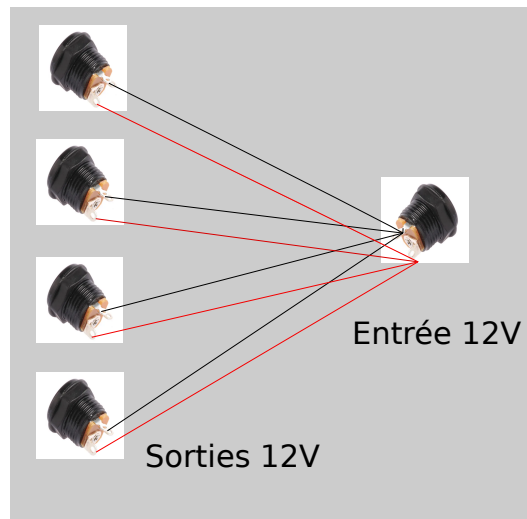


Je me suis fabriqué un boîtier d'alimentation très simple, à partir d'un boîtier de dérivation Legrand© étanche, des prises femelle DC 2,5mm, une prise USB récupéré dans un prise USB allume-cigare. Je dispose donc de 5 sorties 12V, 2 sorties 5V et une entrée 12V alimenté en 220V – 12V-10Amp ou une batterie 12V – xAmp. Je peux ainsi alimenter la monture,

la caméra imageur, le moteur de MAP, la roue à filtre et une résistance chauffante.

J'alimente pour l'instant mes 2 résistances chauffantes avec une autre source.

Le schéma de câblage, en parallèle, est des plus simples. Relié chaque connecteur + de chaque prise femelle au + de l'entrée 12V, de même pour le -. Pour la sortie 5V, un adaptateur 12V – 5V 3A est nécessaire.



5- Une autre solution.

Il est aussi possible de créer un OS à partir de l'OS Raspbian 64 bits Bullseye. Mais pour l'heure, la seule solution pour installer Kstars-Ekos, Indi sera la compilation à partir d'un script. Et c'est long ! De plus pour toute nouvelle version, vous devrez refaire la compilation !

Le script est là: <https://gitea.nouspiro.space/nou/astro-soft-build>.

Installer d'abord les dépendances à l'aide du script *install-dependancies.sh* que vous lancerez dans un terminal par *./install-dependancies.sh*.

Puis vous lancerez la compilation de Kstars, Indi par *./buildsoft-stable.sh*. Allez prendre plusieurs café en attendant que ça se passe !

Pour PHD2 il faut aussi passer par la compilation. Voir le site: <https://github.com/OpenPHDGuiding/phd2/wiki/BuildingPHD2OnLinux>. Et c'est long !

APPENDICE

Script pour mettre à jour la date et heure système à partir d'un GPS. Créez un fichier **Update_date.sh** avec le contenu suivant, sauvegardez le dans votre répertoire utilisateur `home/user/`.

```
#!/usr/bin/python
import os
import sys
import time
from gps import *
print 'Set System Clock to GPS UTC time'
try:
    gpsd = gps(mode=WATCH_ENABLE)
except:
    print 'ERROR: No GPS Present, time not set!!'
    sys.exit()
while True:
    #wait until the next GPSD time tick
    gpsd.next()
    if gpsd.utc != None and gpsd.utc != "":
        #gpsd.utc is formatted like "2015-04-01T17:32:04.000Z"
        #convert it to a form the date -u command will accept: "20140401 17:32:04"
        #use python slice notation [start:end] (where end desired end char + 1)
        # gpsd.utc[0:4] is "2015"
        # gpsd.utc[5:7] is "04"
        # gpsd.utc[8:10] is "01"
        gpsutc = gpsd.utc[0:4] + gpsd.utc[5:7] + gpsd.utc[8:10] + ' ' + gpsd.utc[11:19]
        os.system('sudo date -u --set="%s" % gpsutc')
    sys.exit()
```

Pour l'exécuter automatiquement, dans le menu *Système-Préférences-Personnel*, choisissez l'entrée *Applications au démarrage*.



Ajouter une entrée, avec une désignation et la ligne de commande suivante: `sudo sh ~/Update_date.sh`.

Lors du boot, après démarrage de la session graphique, votre programme sera exécuté. Ajouter un délai, le temps que le GPS soit effectif.

