Klipper Firmware en KP3S

**Índice**

* **Guia Detallada de Instalación … 2**
* **Calibrar los PID de temperatura … 4**
* **Calibrar los pasos del Extrusor … 4**
* **Nivelación de la Cama Caliente … 5**
* **Calibrar el Pressure Advance … 7**
* **Calibrar el Flujo de impresión … 7**
* **Eliminar vibraciones y resonancias … 8**
* **Calibrar las Retracciones … 9**
* **Fichero de configuración KP3S.cfg … 10**
* **Configuración con SAMBA … 12**
* **Configuración de KlipperScreen … 12**
* **Configuración pantalla OLED … 13**
* **Algunas Macros y mejoras … 14**
* **Otros Temas y Curiosidades … 17**
* **Configuración TMC2209 … 18**
* **Configuración Beeper … 19**
* **Klipper + Octoprint, método simple … 20**

Guia Rápida de Instalación

Este es un tutorial lo más concreto y escueto posible para la instalación de Klipper en la KP3S. Está reducido al máximo y presupone unos conocimientos básicos respecto a ficheros, conexiones y procesos.

**Material necesario:**

* Raspberry Pi 3/4/400 o Raspberry Pi Zero W *(no aceptará Octoprint en el futuro)*
* Tarjeta MicroSD de 8Gb mínimo *(16Gb. recomendados)*
* Cable USB de Raspberry PI a la impresora
* Grabación de la SD, Raspberry Pi Imager [*https://www.raspberrypi.org/software/*](https://www.raspberrypi.org/software/)
* Cliente de acesso SSH y SFTP: MobaXterm (gratuito) [*https://mobaxterm.mobatek.net*](https://mobaxterm.mobatek.net)
* Archivo de firmware para impresora “robin\_nano.bin” *(compilar o en el grupo)*
* Archivo de configuración de la impresora “KP3S.cfg” *(disponible en el grupo)*

**Pasos a seguir:**

| 1.- Preparar SD con “Raspberry PI OS Lite 32bits” con Programa “Raspberry Imager”  2.- Entrar en los ajustes del Raspberry Pi Imager  3.- Ajustar: Nombre, activar SSH, definir contraseña usuario “pi” y configurar wifi  4.- Introducir la SD en la Raspberry Pi y arrancar  5.- Acceder a la RPi por SSH *(nombre:"nombre.local" user:"pi" y contraseña)*  6.- Actualizar el Sistema operativo: **sudo apt update** y **sudo apt full-upgrade**  7.- Montar GIT: **sudo apt-get install git -y**  8.- Montar KIAUH:  **git clone https://github.com/th33xitus/kiauh.git**  9.- Ejecutar KIAUH: **./kiauh/kiauh.sh**  10.- En KIAUH, I**NSTALL [1]**, Instalar **KLIPPER [1], MOONRAKER [2] y FLUIDD [4]** 11.- Copiar el "**robin\_nano.bin**" *(en el grupo)* en la SD y arrancar con él la KP3S  12.- Cuando arranque y la pantalla quede negra, apagar y retirar SD  13.- Conectar el USB a la Raspberry e impresora, y arrancar  14.- Conectarse por web al Fluidd (**http://raspberrypi.local** o la **http://ip**)  15.- Subir el **KP3S.cfg** a la configuración  16.- Crear el “**printer.cfg**” con los ajustes básicos *(Include + serial + macros + sd)*  17.- Reiniciar Klippper y conectar y Realizar los ajustes y calibraciones pertinentes |
| --- |

En el caso de dudas o querer entrar en detalles precisos y explicaciones, ir a la siguiente sección, donde se amplía toda la información existente.

**Para la instalación rápida de Octoprint + Klipper (Pág.20 y siguientes):**

| 1.- Preparar SD con OctoPi+Klipper con Programa Raspberry Imager  2.- Poner nombre, activar SSH, configurar wifi y contraseña en ajustes, y grabar SD  3.- Introducir la SD en la Raspberry Pi y arrancar  4.- Acceder a la RPi por SSH *(nombre:"raspberrypi.local" user:"pi" pass:"raspberry")*  5.- Actualizar el Sistema operativo: **sudo apt update** y **sudo apt full-upgrade**  6.- Copiar en “/home/pi”, el fichero **printer.cfg**  7.- Copiar el "**robin\_nano.bin**" *(en el grupo)* en la SD y arrancar con él la KP3S  8.- Cuando arranque y la pantalla quede negra, apagar y retirar SD  9.- Conectarse por web al Fluidd (**http://raspberrypi.local** o la **http://ip**)  10.- Reiniciar la Pi y realizar los ajustes y calibraciones pertinentes |
| --- |

Guia Detallada de instalación

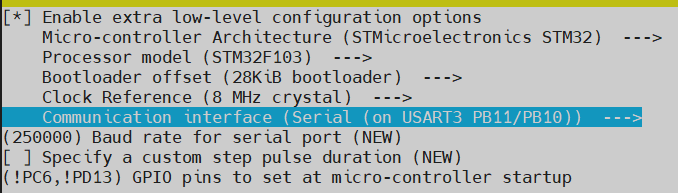
1. Preparar SD con Raspberry PI OS Lite *(puede ser cualquier versión, la Lite lo mínimo)*

[**https://www.raspberrypi.org/software/**](https://www.raspberrypi.org/software/)

1. Poner en los ajustes del Raspberry Pi Imager:  
    *Nombre que se desee para la Pi  
    Activar el SSH  
    Definir una contraseña para el usuario “pi”  
    Configurar la wifi si se va a utilizar*
2. Introducir la SD en la Raspberry Pi y arrancar
3. Acceder a la RPi por SSH *(nombre:"****raspberrypi.local****" user:"****pi"*** *pass:"****raspberry****")  
   Utilizar, por ejemplo, el MobaXterm:* [*https://mobaxterm.mobatek.net*](https://mobaxterm.mobatek.net)
4. Actualizar el Sistema operativo:  **sudo apt update**  y  **sudo apt full-upgrade**
5. Montar GIT:  **sudo apt-get install git -y**
6. Montar KIAUH:  **git clone https://github.com/th33xitus/kiauh.git**
7. Ejecutar KIAUH:  **./kiauh/kiauh.sh**
8. En el KIAUH, ir a **INSTALL** **[1],**  instalar **KLIPPER [1]**, **MOONRAKER [2]** y **FLUIDD [4]**
9. Si **NO** se quiere generar y compilar el firmware de la impresora, saltar al ***Punto 16***
10. Volver al menú principal del KIAUH

Entrar en la opción **ADVANCED [4]** y luego a **Firmware - Build Only [3]**

1. Ajustar los parámetros tal como indica la imagen



*Tener especial cuidado en los datos de: Processor, Bootloader, y USART3*

*Los dos GPIO pins del final, desactivan la pantalla de la impresora*

1. Pulsar la **[Q]** para salir y la **[Y]** para guardar, se compilará el firmware.
2. Salir del KIAUH *([Q] 2 veces)* y ejecutar las órdenes:

**cd ~/klipper   
 ./scripts/update\_mks\_robin.py out/klipper.bin out/robin\_nano.bin**

1. Conectarse por SFTP a la Raspberry, y descargar el fichero “**robin\_nano.bin**” de la carpeta “**/home/pi/klipper/out/**”. *Con el MobaXterm, también.*
2. Copiar el "**robin\_nano.bin**" *(también está compilado en el grupo de Telegram)*, en el directorio raíz de la MicroSD y arrancar con él la KP3S
3. Cuando arranque y la pantalla quede negra (sin textos, tardará pocos segundos en grabarse), apagar y retirar SD, conectar el USB a la Raspberry e impresora, y arrancar
4. Conectarse con web browser al Fluidd ([**http://raspberrypi.local**](http://raspberrypi.local) o http://ip\_raspberry)
5. Subir el **KP3S.cfg** *(en el grupo/doc)* a la configuración *(atención a mayúsc./minúsc.)*
6. Crear o introducir los siguientes datos en el “**printer.cfg**”

| [include KP3S.cfg]  [include kiauh\_macros.cfg]  [mcu]  serial: /dev/serial/by-id/usb-1a86\_USB2.0-Serial-if00-port0  restart\_method: command  [virtual\_sdcard]  path: ~/gcode\_files  [respond]  default\_type: command  [pause\_resume]  [display\_status] |
| --- |

1. Reiniciar el Klipper y esperar que sincronice *(puede tardar hasta 20 segundos)*
2. En el caso de fallo de comunicación con el serial, comprobar con el KIAUH de la siguiente forma:

* Conectar la impresora y las Raspberry por el USB
* Acceder por terminal a la Raspberry PI
* Ejecutar KIAUH:  **./kiauh/kiauh.sh**
* En la opción **Advanced [4] => Get MCU ID [6]**
* Aparecerá el nombre del puerto serie concreto a utilizar en el printer.cfg
* Modificar el “printer.cfg” con el nombre exacto, y reiniciar el klipper

1. *En caso de otros problemas, ir al grupo, reportar los errores y preguntar*

Calibrar PID de temperatura

El fichero KP3S.cfg, que entregamos por defecto, lleva el control de temperatura por “watermark”. Este sistema gestiona la temperatura por los picos de la “ola” que alcanza cuando sube y empieza a bajar. Es efectivo pero poco preciso.

Ejecutar la calibración del PID del extrusor (tiempo estimado, 5 minutos):   
  
 **PID\_CALIBRATE HEATER=extruder TARGET=200**

Ejecutar la calibración del PID de la cama (tiempo estimado, 5 minutos):  
  
 **PID\_CALIBRATE HEATER=heater\_bed TARGET=60**

Editar y sustituir *(“control: watermark”)* en “**KP3S.cfg**” con los datos obtenidos:

| [extruder]  control: pid  pid\_Kp: XX.XXX  pid\_Ki: XX.XXX  pid\_Kd: XX.XXX |  | [heater\_bed]  control: pid  pid\_Kp: XX.XXX  pid\_Ki: XX.XXX  pid\_Kd: XX.XXX |
| --- | --- | --- |

*Se puede utilizar el “save\_config”, pero esté genera bastante basura y desorden en los “cfg”*

Recalibrar pasos del Extrusor

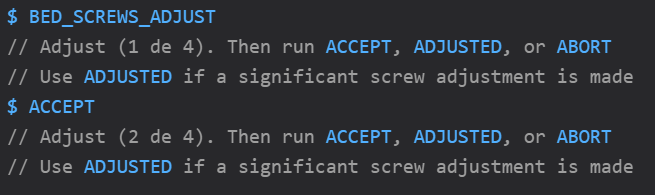
* Asegurarse de que hay filamento en el extrusor, y el hotend está caliente
* Abrir el “**KP3S.cfg**”, y buscar el dato “**rotation\_distance**” dentro de **[extruder]**
* Ese dato que aparece,le llamaremos **<PREVIO>**
* Hacer un par o tres de extrusiones de unos 50 mm de filamento (con el Fluidd)
* Con un rotulador, marcar el filamento a unos 70 mm de la entrada del extrusor
* Medir con un “pie de rey” la distancia hasta la marca. El dato es **<INICIAL>**
* Extruir 50 mm de filamento (**G1 E50 F60**), este dato es: **<REQUERIDOS>**
* Medir desde la entrada del extrusor hasta la marca. Esta será **<RESTANTE>**.
* Entonces calculamos: **<ACTUAL> = <INICIAL> - <RESTANTE>**
* Calcular así: **rotation\_distance = <PREVIO> \* <ACTUAL> / <REQUERIDOS>**
* Redondear sobre el 3r decimal el resultado.
* Modificar en **[extruder]** el “**rotation\_distance**” con el dato calculado.
* con **G91**, reseteamos extrusor, por si debemos extruir de nuevo

Nivelación de Cama Caliente

Klipper intenta en lo posible, la menor interacción y manipulación física de la impresora. La forma de calibrar la cama con él, se basa en esa premisa. Primero se debe realizar la **nivelación básica,** y posteriormente la **manual o automática** según se disponga.

Es importante haber **calentado previamente la cama y el hotend** para una correcta alineación.

**Nivelación BÁSICA** de la cama por las “roscas de ajuste” de la misma:

 Añadir al cfg de la configuración de la impresora, la siguiente sección:

| [bed\_screws]  screw1: 20, 20  screw1\_name: (1 de 4)  screw2: 160, 20  screw2\_name: (2 de 4)  screw3: 160, 160  screw3\_name: (3 de 4)  screw4: 20, 160  screw4\_name: (4 de 4) horizontal\_move\_z: 10  speed: 250 |
| --- |

Introducir la siguiente orden en la consola:

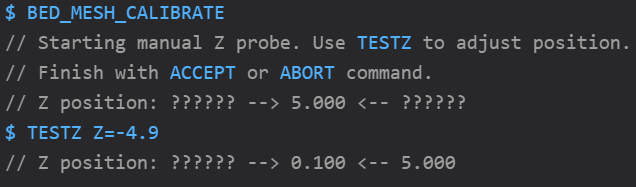
**BED\_SCREWS\_ADJUST**

Y seguir las instrucciones de la consola:

* **ACCEPT** *(confirmar) /* **ADJUSTED** *(se ha tocado la rosca manualmente)*
* **ABORT** *(cancela el nivelado)*

en cada uno de los puntos

**Nivelación MANUAL por MALLA** de la cama (9 puntos):

 Añadir al cfg de la configuración de la impresora, la siguiente sección:

| [bed\_mesh]  speed: 800 *# Velocidad entre puntos*  mesh\_min: 10, 10 *# Posición inicial*  mesh\_max: 170, 170 *# Posición final*  algorithm: lagrange *# Cálculo interpolación*  probe\_count: 3, 3 *# Puntos en X/Y*  mesh\_pps: 2,2 *# Sub-malla cálculo*  split\_delta\_z: .010 *# Precisión en Z*  fade\_end: 10 *# Distancia en Z finaliza* |
| --- |

En los ajustes del Fluidd, hacer el **“Home ALL”**, y luego **“Calibración”**, y seguir las instrucciones de la **CÓNSOLA** del Fluidd

El comando para el ajuste de la Z es:

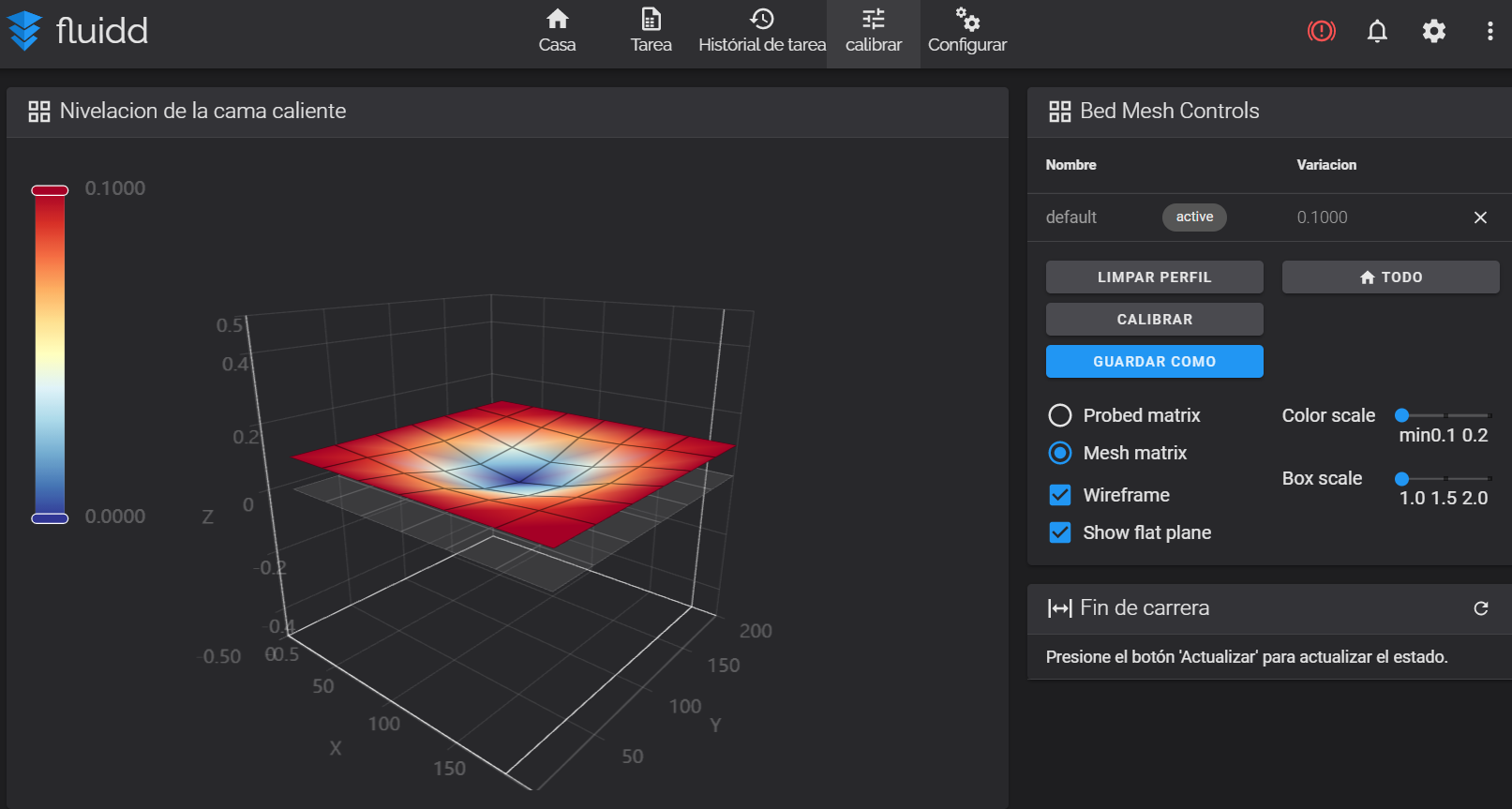
**TESTZ Z=*<offset>***

Donde *<offset>* es el desplazamiento respecto a la cifra actual (5 mm por defecto). Es decir, si se quiere ir hasta la posición Z=0, será: **TESTZ Z=-5**

Los ajustes con los “babysteps”, funcionan, se ejecutan y guardan correctamente.

Hacer el **ACCEPT / ABORT** *(confirma ajuste y pasa a siguiente)* según convenga

En el caso de querer añadir un perfil de calibración manual por defecto:



| [bed\_mesh default]  version: 1  points: 0.05, 0.05, 0.05  0.05, 0.00, 0.05  0.05, 0.05, 0.05  tension: 0.2  y\_count: 3  min\_y: 10  max\_y: 170  mesh\_y\_pps: 2  x\_count: 3  min\_x: 10  max\_x: 170  mesh\_x\_pps: 2  algo: lagrange |
| --- |

**Nivelación AUTOMÁTICA** con BL-Touch (o compatibles):

* Hay que añadir la parte de la nivelación MANUAL por malla [bed\_mesh] del punto anterior (no es necesario el perfil por defecto)

| [stepper\_z]  endstop\_pin: probe:z\_virtual\_endstop *# SUSTITUIR o COMENTAR PREVIO para BLTouch*  [safe\_z\_home]  home\_xy\_position: 90,90 *# SUSTITUIR o COMENTAR PREVIO para BLTouch*  [bltouch]  sensor\_pin: ^PA11  control\_pin: PA8  x\_offset: -29  y\_offset: -1  z\_offset: 2.1 *# Offset respecto a la cama*  speed: 10  samples: 1  samples\_result: average  stow\_on\_each\_sample: false  pin\_up\_touch\_mode\_reports\_triggered: false  *#probe\_with\_touch\_mode: True # Activar para BL-Touch v3.0 ORIGINAL*  *#set\_output\_mode: 5V* |
| --- |

Ir a la pantalla de ajustes del Fluidd, hacer el home, y luego la calibración, al finalizar, realizar el save\_config.

Atención: cuidado con lanzar muchas pruebas seguidas con el BL-Touch, pq es bastante delicado y puede fallar, y se detiene todo el proceso con un error extraño. Si las pruebas se van fuera de su margen de tolerancia preajustado, se detiene por seguridad.

Calibrar el Pressure Advance

Los parámetros entre lo que oscila el PA son:

* Extrusión directa: de 0.0 a 0.1 (0.02 a 0.05 habitualmente)
* Extrusión con bowden: de 0.0 a 1.0 ( 0.4 a 0.6 habitualmente)

La velocidad afecta mucho al PA y adherencia de las capas, en caso de baja adherencia, aumentar temperatura de salida (5ºC a 10ªC)

Hay muchos métodos para afinar el PA, pero cómo intervienen tanto factores, los oficiales son lentos y tampoco reflejan la realidad (puede variar por muchos factores)

Mi consejo es poner a imprimir uno de los famosos cubos de calibración de unos 4x4 o 5x5 cm., hueco por dentro (o relleno cero), y cada 10 a 20 capas (2-4 mm de altura), hacer un pause y observar, e incrementar el PA en un par de unidades (0.02 o 0.2, dependiendo del extrusor), y así sucesivamente, hasta llegar al factor máximo.

**SET\_PRESSURE\_ADVANCE ADVANCE=0.02**

Entonces, despegar la pieza, observar las esquinas del cubo, y la costura (cambios en Z) de la pieza, y decidir qué esquinas quedan menos “abultadas”, y la costura lisa y precisa.

Tras pruebas similares, obtendréis un PA de referencia para vuestra impresora.

Se puede registrar en el fichero de configuración de la impresora:

| [extruder]  pressure\_advance: 0.04 |
| --- |

Calibrar Flujo de impresión

Descargarse el “**flow\_calibrator.stl**” del grupo de Telegram.

Ajustar los parámetros por defecto de la impresora con velocidad y flujo al 100%

Es importante haber calibrado el Pressure Advance previamente:

* Si la costura no queda lisa, (sobresale o se hunde), ajustar el PA para que quede correcta tanto en el anillo como en el cilindro (interna y externa)
* Si aparece “pata de elefante”, la boquilla está muy pegada a la cama, ajustar de nuevo.

Cuando la impresión sea correcta, introducir las dos piezas una dentro de la otra:

* Si las dos no encajan, bajar el flujo un 10%. Si entran muy sueltas, subir el flujo un 10%. Repetir la prueba hasta conseguir el resultado deseado.
* Cuando las dos encajen, ajustar el flujo (2%-3%) arriba y abajo de tal forma que entren, sin hacer una fuerza excesiva, como si se deslizaran, con una leve presión (no holgado)

Eliminar Resonancias

Material necesario:

* Placa con chip ADXL345  
  *Importante: forrarla con cinta aislante (evitar cruces) y recordar la orientación chip en X*

Antes de todo, la impresora debe estar en una superficie fija y estable, que no esté sometida a vibraciones por su soporte, sino, afianzar, trabarla, o simplemente ponerla en el suelo

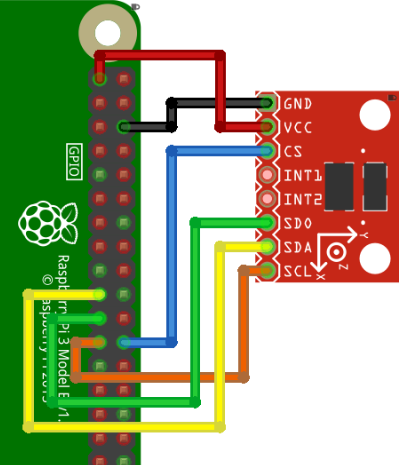
Instalar el "numpy" (librerías matemáticas) y sus librerías de dibujo.

**~/klippy-env/bin/pip install -v numpy**

**sudo apt install python-numpy python-matplotlib**

Entrar en la configuración de la Raspberry:  **sudo raspi-config**

Activar el interfaz SPI yendo a: **[3] Interface Options => [P4] SPI Enable**

Añadir esto al **“printer.cfg”** y conectar según el esquema

| [mcu rpi]  serial: /tmp/klipper\_host\_mcu  [adxl345]  cs\_pin: rpi:None  [resonance\_tester]  accel\_chip: adxl345  probe\_points: 90,90,50 |
| --- |

Cambiar el límite de aceleración y deceleración de la impresora:

| [printer]  max\_accel: 10000  max\_accel\_to\_decel: 10000 |
| --- |

Cuando este todo conectado y sin errores, ejecutar en el Klipper:

**ACCELEROMETER\_QUERY**

Comprobar que nos devuelva valores y no se queje *(no importan valores y signos)*

Colocar bien alineado *(sobre el extrusor, alineado respecto al eje de la X del brazo)*, y bien fijado *(con cinta aislante, carrocero o similar)*, y ejecutar el test (aprox 1-2 min.):

**TEST\_RESONANCES AXIS=X**

Cambiar el acelerómetro de ubicación y colocar enganchado a la cama EN LA MISMA POSICIÓN que lo teníamos en el brazo (así estaremos alineando ya la Y), y ejecutar:

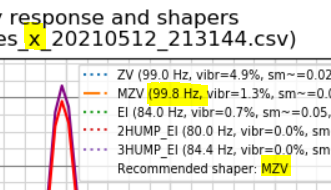
**TEST\_RESONANCES AXIS=Y**

Al finalizar las pruebas, ejecutar los scripts:

**~/klipper/scripts/calibrate\_shaper.py /tmp/resonances\_x\_\*.csv -o /tmp/shaper\_calibrate\_x.png**

**~/klipper/scripts/calibrate\_shaper.py /tmp/resonances\_y\_\*.csv -o /tmp/shaper\_calibrate\_y.png**

Conectarse por SFTP (mobaxterm) y descargarse los archivos PNG de la ruta “/tmp”

Establecer los parámetros en la configuración de la impresora, según los resultados (ejemplo):

|  | [input\_shaper]  shaper\_type\_x: mzv  shaper\_freq\_x: 99.8  shaper\_type\_y: ei  shaper\_freq\_y: 50.8 |  |
| --- | --- | --- |

**Parece que en la última versión de klipper/raspiimage ha cambiado la configuración y hay que hacer más cosas, como activar el mcu**

<https://www.klipper3d.org/RPi_microcontroller.html>

After installing Klipper, install the script. run:

**cd ~/klipper/**

**sudo cp "./scripts/klipper-mcu-start.sh" /etc/init.d/klipper\_mcu**

**sudo update-rc.d klipper\_mcu defaults**

To compile the Klipper microcontroller code, start by configuring it for the "Linux process":

**cd ~/klipper/**

**make menuconfig**

In the menu, set "Microcontroller Architecture" to "Linux process," then save and exit.

To build and install the new microcontroller code, run:

**sudo service klipper stop**

**make flash**

**sudo service klipper start**

If klippy.log reports a "Permission denied" error when attempting to connect to /tmp/klipper\_host\_mcu then you need to add your user to the tty group. The following command will add the "pi" user to the tty group:

**sudo usermod -a -G tty pi**

Calibrar las Retracciones

**AYUDA!!!**

**A ver a quién se le ocurre una forma simple de poder ajustar las retracciones, y lo plasmamos aquí.**

Retracciones:

en directo, entre 0.1 mm y 3 mm habitualmente

con bowden, entre 3 mm y 8 mm habitualmente

La velocidad de impresión puede afectar a las retracciones

Sobre el ajuste z-hop

En el caso de querer definir las retracciones en la propia impresora, añadir:

| [firmware\_retraction]  retract\_length: 0.9 *# Retracción en mm (Bowden=3.2 directa=0.9)*  retract\_speed: 35  unretract\_speed: 35  unretract\_extra\_length: 0 *# en el caso de ser necesario un cebado* |
| --- |

Para que las retracciones funciones por hardware (códigos G10 y G11), es necesario realizar ajustes en los laminadores:

* PrusaSlicer
* Ultimaker Cura
* IdeaMaker

**SET\_RETRACTION [RETRACT\_LENGTH=<mm>] [RETRACT\_SPEED=<mm/s>] [UNRETRACT\_EXTRA\_LENGTH=<mm>] [UNRETRACT\_SPEED=<mm/s>]**

Fichero configuración KP3S.CFG

| [printer]  kinematics: cartesian  max\_velocity: 300  max\_accel: 3000  max\_accel\_to\_decel: 3000  max\_z\_velocity: 25  max\_z\_accel: 100  *########## PARÁMETROS ESPECIALES ######*  [static\_digital\_output display\_reset]  pins: !PC6, !PD13  [gcode\_arcs]  resolution: 0.1  *############### HOMING ################*  [safe\_z\_home]  home\_xy\_position: 0,0  speed: 50  z\_hop: 10  z\_hop\_speed: 15  *############# STEPPERS ################*  [stepper\_x]  step\_pin: PE3  dir\_pin: PE2  enable\_pin: !PE4  microsteps: 32  rotation\_distance: 40  endstop\_pin: ^!PA15  position\_endstop: 0  position\_min: 0  position\_max: 180  homing\_speed: 50  homing\_retract\_dist: 0  [stepper\_y]  step\_pin: PE0  dir\_pin: PB9  enable\_pin: !PE1  microsteps: 32  rotation\_distance: 40  endstop\_pin: ^!PA12  position\_endstop: 0  position\_min: 0  position\_max: 180  homing\_speed: 50  homing\_retract\_dist: 0 |  | [stepper\_z]  step\_pin: PB5  dir\_pin: !PB4  enable\_pin: !PB8  microsteps: 32  rotation\_distance: 8  endstop\_pin: !PA11  position\_endstop: 0  position\_min: -1  position\_max: 180  homing\_speed: 10  homing\_retract\_dist: 3.0  *############# EXTRUSOR ##############*  [extruder]  step\_pin: PD6  dir\_pin: !PD3  enable\_pin: !PB3  microsteps: 32  nozzle\_diameter: 0.400  filament\_diameter: 1.750  max\_extrude\_only\_distance: 150.0  max\_extrude\_cross\_section: 50.0  heater\_pin: PC3  sensor\_type: ATC Semitec 104GT-2  sensor\_pin: PC1  min\_temp: 0  max\_temp: 250 *#Titan:7.1 Bondtech:7.755*  rotation\_distance: 33.54  pressure\_advance: 0.00  control: watermark  *############## CAMA CALIENTE ########*  [heater\_bed]  heater\_pin: PA0  sensor\_type: EPCOS 100K B57560G104F  sensor\_pin: PC0  min\_temp: 0  max\_temp: 130  control: watermark  *############### VENTILADORES ########*  [fan]  pin: PB1  [heater\_fan my\_nozzle\_fan]  pin: PB0  heater: extruder  heater\_temp: 50.0  fan\_speed: 1.0  shutdown\_speed: 1 |
| --- | --- | --- |

Configuración con SAMBA

SAMBA es nombre que se le da al sistema de intercambio de archivos por la red de Windows

* Instalar el paquete del Samba para linux

**sudo apt-get install samba samba-common-bin**

* Modificar el archivo de configuración para añadir la carpeta a compartir

**sudo nano /etc/samba/smb.conf**

Añadir lo siguiente al final del archivo:

| [Klipper\_gcode]  path = /home/pi/gcode\_files  writeable = yes  ; browseable = yes  guest ok = yes  force user = pi  create mask = 0777  directory mask = 0777 |
| --- |

*Para salir del editor,* ***ctrl+X****, luego* ***Y****, confirmar el nombre del archivo.*

* Resetear el servicio Samba

**sudo systemctl restart smbd**

* Para Acceder a la carpeta, desde windows: **\\ip.de.la.raspberry\Klipper\_gcode**

allí podréis ver los archivos contenidos, borrar y subir nuevos

Configuración KlipperScreen

Para el conexionado y control de la impresora desde la misma (o a distancia), con una pantalla integrada, estos son los materiales y pasos necesarios:

* Materiales:

La pantalla que tengo es de 5", y es ésta. Se vende bajo muchas marcas distintas, con distintos precios, pero la placa es siempre la misma.

* Instalación y ajustes:

Para hacer funcionar la pantalla se utilizan los siguientes comandos:

**sudo rm -rf LCD-show**

**git clone https://github.com/goodtft/LCD-show.git**

**chmod -R 755 LCD-show**

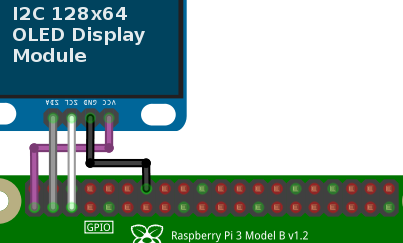
**cd LCD-show/**

**sudo ./LCD5-show**

Para montar el Klipperscreen, ir al menú de KIAUH e instalar.

Configuración Pantalla OLED

En el caso de querer integrar una pantalla de control (solo visión) de forma directa a la Rpi:



* Conectar el cableado de la pantalla según el siguiente gràfico
* Montar libreria, activar el bus I2C en la Raspberry PI y conceder permisos

**sudo apt-get install gpiod**

**sudo raspi-config** en “**[3] Interface Options” => “[P5] I2C” => “Enable”**

**sudo usermod -a -G tty pi**

* Paramos el servicio del Klipper para trabajar sin interferencias

**sudo service klipper-1 stop**

* Establecer el script de inicio y nivel de ejecución

**cd ~/klipper/**

**sudo cp "./scripts/klipper-mcu-start.sh" /etc/init.d/klipper\_mcu**

**sudo update-rc.d klipper\_mcu defaults**

* Generamos el firmware específico de la Raspberry Pi para el control de la pantalla

**make menuconfig**

escoger “**Microcontroller Architecture” => “Linux Process”** y guardar **[Q]**

**make flash**

* Y reiniciamos el servicio y la Pi para que esté todo en su sitio

**sudo service klipper-1 start**

**sudo reboot**

* Añadir o Modificar en el “printer.cfg” o en el fichero de configuración pertinente:

| [mcu host]  serial: /tmp/klipper\_host\_mcu  [display]  lct\_type: ssd1306  menu\_timeout: 30  i2c\_mcu: host  i2c\_bus: i2c.1 |
| --- |

Algunas Macros y mejoras

El M600 es el “PAUSE” de Marlin

| [gcode\_macro M600]  gcode:  {% if printer.extruder.can\_extrude == 1 %}  {% if printer.pause\_resume.is\_paused == 0 %}  PAUSE  {% endif %}  M118 Primero UNLOAD, entonces LOAD y RESUME cuando este listo.  {% else %}  RESPOND TYPE=error MSG="Extrude below minimum temp"  {% endif %}  [gcode\_macro UNLOAD]  gcode:  {% if printer.pause\_resume.is\_paused == 1 %}  {% if printer.extruder.can\_extrude == 1 %}  G91  G1 E5 F300  G1 E-50 F300  G90  {% else %}  RESPOND TYPE=error MSG=""Extrusión por debajo de temperatura mínima"  {% endif %}  {% else %}  RESPOND TYPE=error MSG="La impresora debe estar en PAUSE"  {% endif %}  [gcode\_macro LOAD]  gcode:  {% if printer.pause\_resume.is\_paused == 1 %}  {% if printer.extruder.can\_extrude == 1 %}  G91  G1 E30 F300  G90  {% else %}  RESPOND TYPE=error MSG="Extrusión por debajo de temperatura mínima"  {% endif %}  {% else %}  RESPOND TYPE=error MSG="La impresora debe estar en PAUSE"  {% endif %} |
| --- |

El M900 es similar al Linear Advance de Marlin *(permite test compatible con linear advance)*

| [gcode\_macro M900]  gcode:  {% if 'K' in params %}  {% if 'E' in params %}  SET\_PRESSURE\_ADVANCE EXTRUDER={params.E} ADVANCE={params.K}  {% else %}  SET\_PRESSURE\_ADVANCE ADVANCE={params.K}  {% endif %}  {% endif %} |
| --- |

En caso de perfiles genéricos en los slicer, añadir START\_GCODE, y END\_GCODE al mismo:

| [gcode\_macro START\_GCODE]  gcode:  {% set BED\_TEMP = params.BED\_TEMP|default(60)|float %}  {% set EXTRUDER\_TEMP = params.EXTRUDER\_TEMP|default(190)|float %}  {% set ENDY = printer.toolhead.axis\_maximum.y|float - 50 %}  M140 S{BED\_TEMP} *# Calentamos la cama*  G90 *# Coordenadas absolutas*  G28 *# Hacemos el HOME en XYZ*  M190 S{BED\_TEMP} *# Esperamos que la cama alcance la temp*  M109 S{EXTRUDER\_TEMP} *# Definimos y esperamos temp del extrusor*  G92 E0 *# Reset del extruder*  G1 X2 Y20 Z0.3 F5000 *# Movemos a la posición inicial*  G1 X2 Y{ENDY} Z0.3 F1500 E15 *# dibujamos la 1ª línea (fina)*  G1 X2 Y{ENDY} Z0.4 F5000 *# nos desplazamos levemente*  G1 X2 Y20 Z0.4 F1500 E30 *# dibujamos la 2º línea (gruesa)*  G92 E0 *# reseteamos el extrusor*  G1 Z1.0 F3000 *# movemos en Z para evitar rascadas en la base*  [gcode\_macro END\_GCODE]  gcode:  {% set YMAX = printer.toolhead.axis\_maximum.y|float %}  G91 *# Coordenadas relativas*  G1 X-2 Y-2 F300 E-3 *# Me aparto levemente y hago retract*  G1 Z10 F3000 *# Subo 10mm*  G90 *# Coordenadas absolutas*  G92 E0 *# reseteamos el extrusor*  G1 X0 Y{YMAX} F1000 *# Cabezal al fondo izquierdo*  M140 S0 *# Apaga la cama*  M104 S0 *# Apaga el extrusor*  M106 S0 *# Apaga el ventilador*  M84 *# Desconecta los motores* |
| --- |

| [gcode\_macro CANCEL\_PRINT]  rename\_existing: BASE\_CANCEL\_PRINT  gcode:  {% set ENDY = printer.toolhead.axis\_maximum.y|float - 10 %}  G91 *# Posición relativa*  G1 E-1 F300 *# Retracción filamento quita presión*  G1 Z1 E-1 F6000 *# Levantamos y desplazamos*  G90 *# Posición abosoluta*  G92 E0 *# reseteamos el extrusor*  G1 X0 Y{ENDY} *# Sacamos la bandeja*  M84 *# Apagamos motores*  M106 S0 *# Apaga ventilador capa*  TURN\_OFF\_HEATERS *# Apaga hotend*  CLEAR\_PAUSE  SDCARD\_RESET\_FILE  BASE\_CANCEL\_PRINT |
| --- |

| [gcode\_macro END\_PRINT]  gcode:  M140 S0 *#*  M104 S0 *#*  M106 S0 *#*  G91 *# Posición relativa*  G1 X0 Y0 E-3 F300 *# Retracción filamento quita presión*  G1 Z20 F3000 *# Levantamos y desplazamos*  G90 *# Posición abosoluta*  M84 *# Apagamos motores* |
| --- |

[gcode\_macro G29]

gcode:

{% set BED\_MESH\_PROFILE\_NAME = params.BED\_MESH\_PROFILE\_NAME|default("default") %}

{% if printer.idle\_timeout.state == "Printing" %}

{action\_respond\_info("This command cannot be used while printing")}

{% elif printer.toolhead.homed\_axes != "xyz" %}

G28

{% endif %}

BED\_MESH\_PROFILE REMOVE={BED\_MESH\_PROFILE\_NAME}

BED\_MESH\_CALIBRATE SAVE={BED\_MESH\_PROFILE\_NAME}

SAVE\_CONFIG

Otros Temas y Curiosidades

* Se puede montar Octoprint y Fluidd en el mismo dispositivo, conviven perfectamente.
* Hay una forma de trabajar con Fluidd sin necesidad de montarlo *(se ha de montar Klipper y Moonraker)*: Acceder a la web [**http://app.fluidd.xyz**](http://app.fluidd.xyz) y añadir la impresora a través de la conexión de Moonraker. La configuración queda almacenada en las cookies del navegador, así que es segura. Esto sólo funciona conectado desde la propia red interna, no desde el exterior.
* Se puede mantener un control de versiones (por si hay que retroceder por una actualización problemática) a través del propio KIAUH.  
  **[2] Update => [0] [Enable] Backups before updating**
* Si aparecen errores de texto en los botones o similar, es muy posible que sea un error en el idioma. Acceder a la configuración y forzar el idioma Inglés, por ejemplo.
* Si se desea retornar a Marlin u otras configuraciones en un momento determinado, no hay ningún problema o afectación.

Simplemente insertar el firmware en la impresora, reinicializar y ya está.

* Se puede cambiar el nombre de la raspberry a través del “raspi-config” o el KIAUH  
  **[4] Advanced => [8] Change Hostname**
* Una sola Raspberry puede gestionar varias impresoras simultáneamente. El KIAUH facilita este proceso de instalación. Cada impresora tiene sus ajustes y parámetros independientes. El mayor problema reside en la forma de definir que USB se va a utilizar para cada una de ellas. En caso de llevarlo a cabo, preguntar en el grupo de Telegram experiencias anteriores de usuarios para conocer detalles y técnicas.
* La lógica de trabajo de Klipper y Fluidd es:

· Klipper es el firmware de la impresora en la Raspberry, dispone de una API

· Moonraker, usa la API de Klipper, y monta un Websocket como interfaz

· Fluidd es un servidor web, que se comunica con Moonraker

· Octoprint es un servidor web nativo, que se comunica virtualmente con la API de Klipper

* Para el tema pantallas en la impresora, el soporte actual de Klipper es muy limitado.  
  Hay soluciones para las LCD tipo 12864 (y emuladas), SSD1306 (oled), st7920, etc...

La KlipperScreen y OctoScreen, no deja de ser la salida standard de pantalla de la raspberry, con un autologin, y supervisión del ratón.

Configuración TMC2209

| [tmc2209 stepper\_x]  uart\_pin: PA3  diag\_pin: PA15  uart\_address: 0  run\_current: 0.650  hold\_current: 0.6  driver\_SGTHRS: 125 *# Sensibilidad Home*  stealthchop\_threshold: 9999  interpolate: true  sense\_resistor: 0.110  driver\_TBL: 2  driver\_TOFF: 3  driver\_HEND: 3  driver\_HSTRT: 0  driver\_PWM\_GRAD: 8  driver\_PWM\_LIM: 9  [tmc2209 stepper\_y]  uart\_pin: PA6  diag\_pin: PA12  uart\_address: 0  run\_current: 0.650  hold\_current: 0.6  driver\_SGTHRS: 125 *# Sensibilidad Home*  stealthchop\_threshold: 9999  interpolate: true  sense\_resistor: 0.110  driver\_TBL: 2  driver\_TOFF: 3  driver\_HEND: 3  driver\_HSTRT: 0  driver\_PWM\_GRAD: 8  driver\_PWM\_LIM: 9 |  | [tmc2209 stepper\_z]  uart\_pin: PA1  uart\_address: 0  run\_current: 0.650  hold\_current: 0.6  stealthchop\_threshold: 9999  interpolate: true  sense\_resistor: 0.110  driver\_TBL: 0  driver\_TOFF: 3  driver\_HEND: 2  driver\_HSTRT: 0  driver\_PWM\_GRAD: 8  driver\_PWM\_LIM: 10  [tmc2209 extruder]  uart\_pin: PE5  uart\_address: 0  run\_current: 0.500  hold\_current: 0.250  stealthchop\_threshold: 0  interpolate: false  sense\_resistor: 0.110 |
| --- | --- | --- |

Configuración del Beeper

El M300 es el “Beeper” de Marlin

######################################################################

# M300 : Play tone. Beeper support

# M300 [P<ms>] [S<Hz>] # P is the tone duration, S the tone frequency.

| [output\_pin BEEPER\_pin]  pin: PC5 *# Beeper pin. This parameter must be provided.*  pwm: true *# A piezo beeper needs a PWM signal, a DC buzzer doesn't.*  value: 0 *# Silent at power on, set to 1 if active low.*  shutdown\_value: 0 *# Disable at emergency shutdown*  cycle\_time: 0.001 *# PWM frequency: 0.001=1ms will give a base tone of 1kHz*  scale: 2000 *# PWM will be in the range of (0-2000 Hz).*  [gcode\_macro M300]  gcode:  {% set S = params.S|default(1500)|int %} *# Por defecto 1 kHz (Frecuencia)*  {% set P = params.P|default(100)|int %} *# Por defecto 100 ms (Duración)*  SET\_PIN PIN=BEEPER\_pin VALUE={S}  G4 P{P}  SET\_PIN PIN=BEEPER\_pin VALUE=0 |
| --- |

Klipper + OctoPrint en KP3S

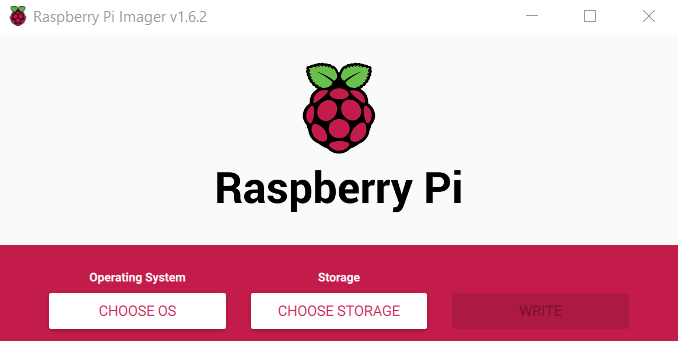
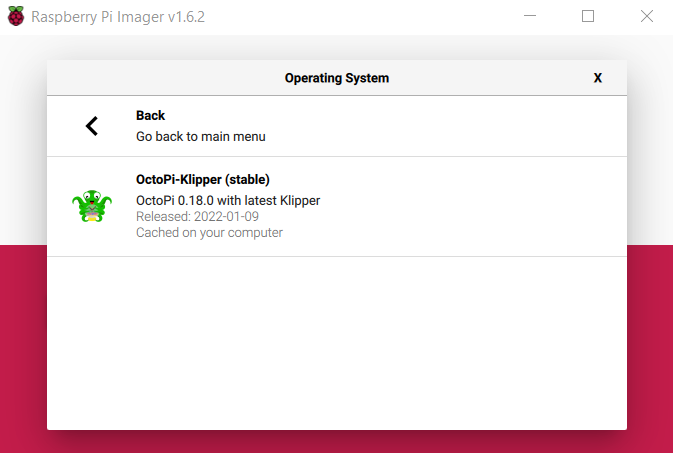
*Instalación mínima necesaria: Raspberry Pi Zero 2 W, Pi 3 o superiores*

*Solo se permite una instancia, no es necesario moonraker, fluidd, etc…*

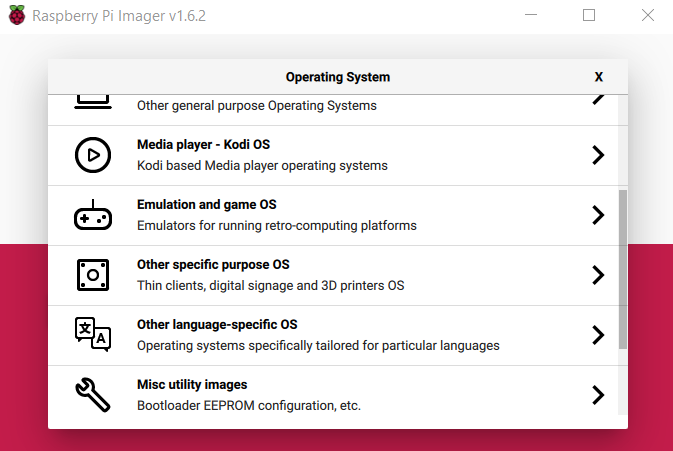
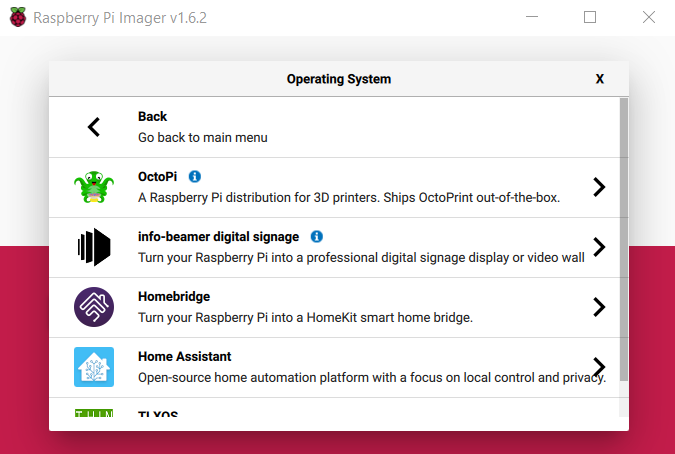
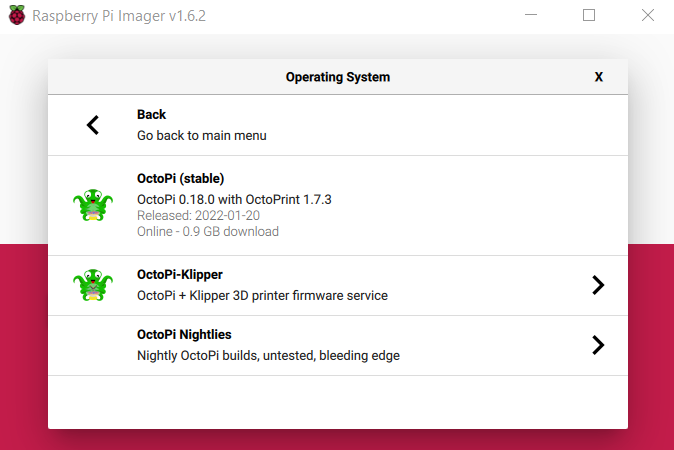
*Se pueden añadir mejoras, aunque se necesitan algunos conocimientos avanzados*

Con el Programa “Raspberry Pi Imager”, grabar la imagen **“OctoPi +Klipper”**

[**https://www.raspberrypi.com/software/**](https://www.raspberrypi.com/software/)

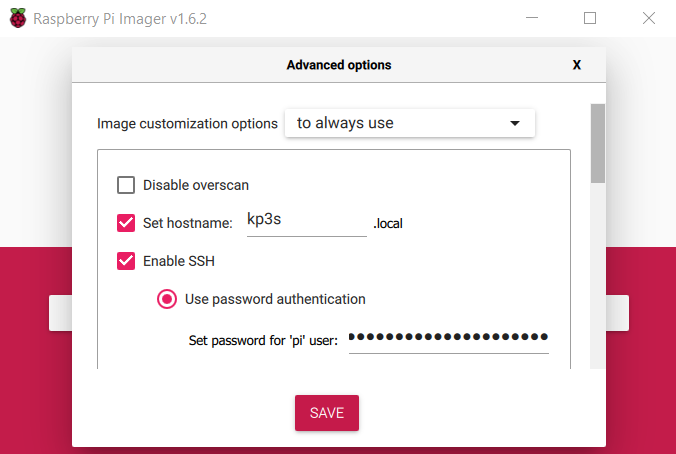
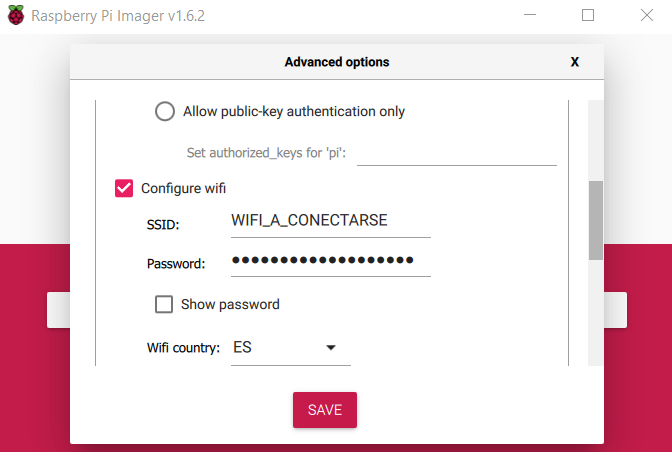
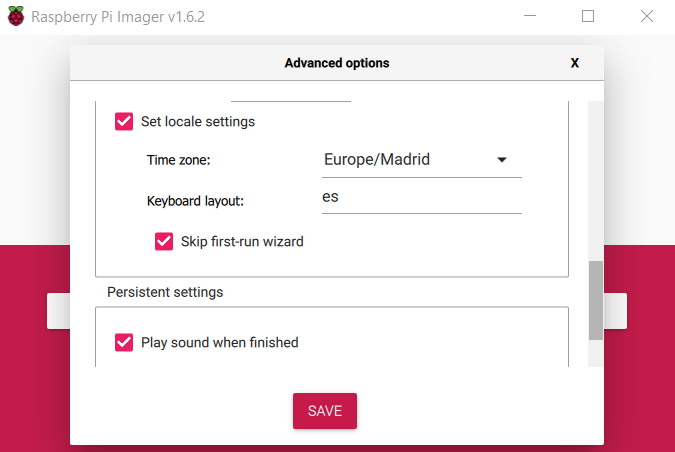
 

Disponible dentro de **“Other Specific Purpose OS”**

Al haber hecho la selección del sistema operativo, pulsar **“ctrl+shift+x”**

**Activar el SSH** y el resto de opciones necesarias *(wifi, localización, …)*

Seleccionar la SD para grabación y proceder a la misma.

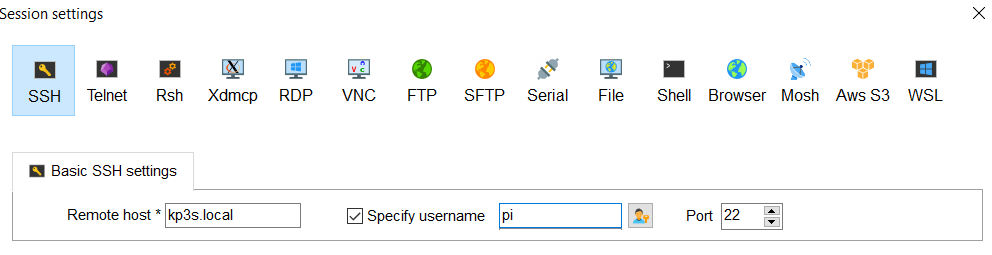
Con la tarjeta SD grabada y verificada, insertarla en la Raspberry Pi

Al iniciarla, esperar un par de minutos para que realice todos los procesos

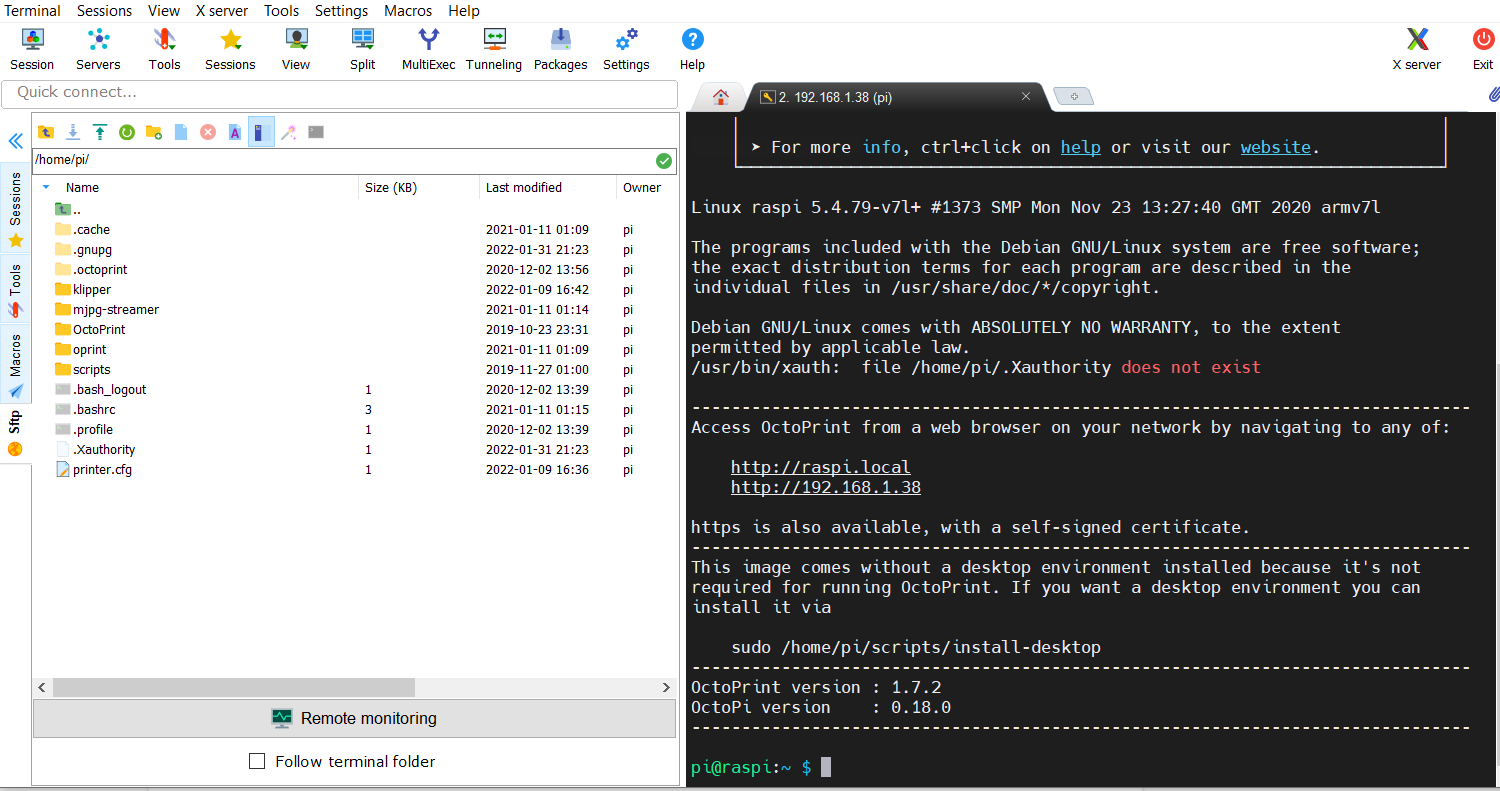
Para acceder a la misma, con el programa **MobaXTerm** (o similar)

[**https://mobaxterm.mobatek.net/download.html**](https://mobaxterm.mobatek.net/download.html)

Conectarse por SSH contra la Rpi con el nombre definido y usuario por defecto “pi”



Introducir la contraseña cuando la pida y obtendremos una pantalla similar a esta:



A la izquierda, depositar el fichero en carpeta **“/home/pi”**

*(arrastrar y soltar con el MobaXterm)*, **“printer.cfg”** de la impresora

*(disponible en: http://…………. )*

*Incluye printer.cfg, robin\_nano.bin (firmware) y octokp3s.zip (backup Octoprint)*

Escribir el terminal *(introducir contraseña y confirmar cuando solicite)*:

**sudo apt update**

**sudo apt full-upgrade**

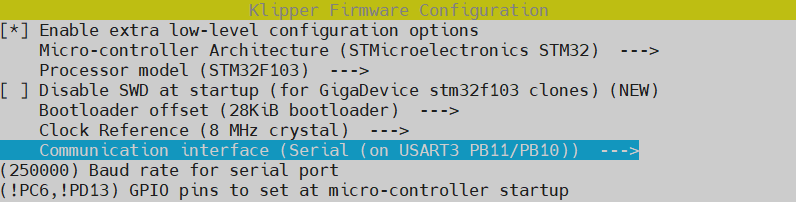
En el caso de necesitar grabar el firmware de la impresora, puede grabar fichero **“Robin\_nano.bin”** en una SD formateada en FAT, insertarlo en impresora y arrancarla.

Si se desea generar nativamente el firmware y grabar el firmware en la KP3S:

**cd klipper**

**make menuconfig**

y ajustar los parámetros según los siguientes ajustes:



Pulsar la [Q] para salir e [Y] para guardar. Para compilar, escribir: **make**

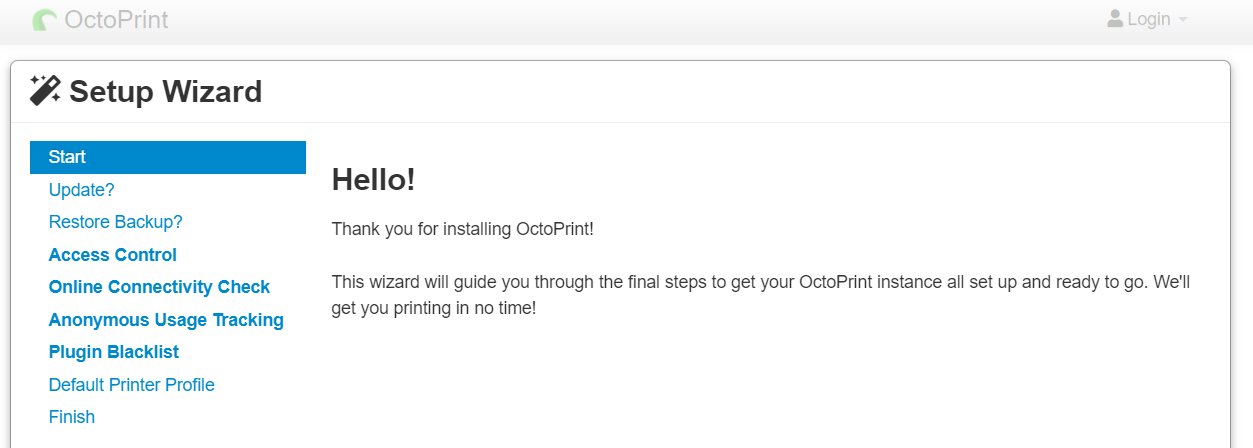
Posteriormente, para adaptar al formato de la KP3s, ejecutar:

**./scripts/update\_mks\_robin.py out/klipper.bin out/Robin\_nano.bin**

Con el MobaXterm, descargar **“robin\_nano.bin”** de **“/home/pi/klipper/out/”**

Después de reiniciar la PI *(“sudo reboot” o manualmente)*, acceder al octoprint:

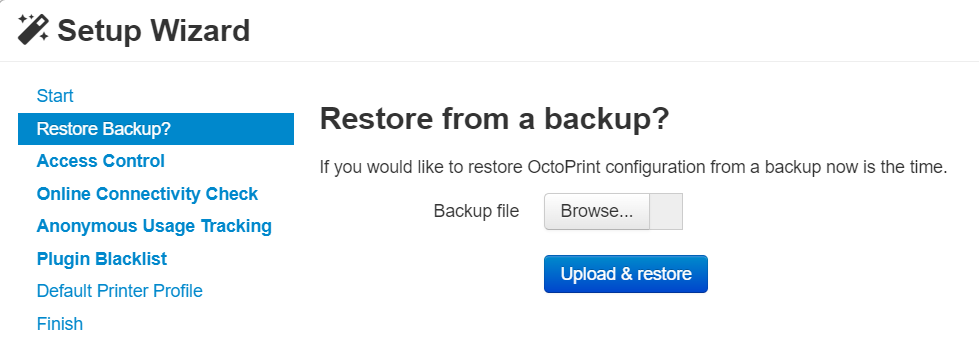
[**http://kp3s.local**](http://kp3s.local) *(o el nombre que se haya definido previamente, o su ip)*



Pulsar abajo a la derecha en **Next**, y seguir el proceso.

En el caso de que indique que hay una actualización, proceder a instalarla y recargar.

Si se desea utilizar el perfil prediseñado por el grupo de usuarios, en la restauración:



Seleccionar el fichero de copia de seguridad del OctoPrint del grupo, y cargarlo.

Acceder de nuevo, e introducir el usuario y contraseña predefinidos en la restauración:

*(Estos datos de entrada pueden ser reajustados posteriormente)*

usuario: **pi**

contraseña: **raspberry**

