

## Manual para utilizar os pinos GPIOs da placa MKS Pi V1.0

Uma das funcionalidades do GPIOs de uma placa Pi (MKS, Raspberry, Orange, etc.) é poder acionar dispositivos ou ler sensores, sem que seja necessário o uso da placa escrava, no caso de se utilizar o Klipper e também do Octoprint.

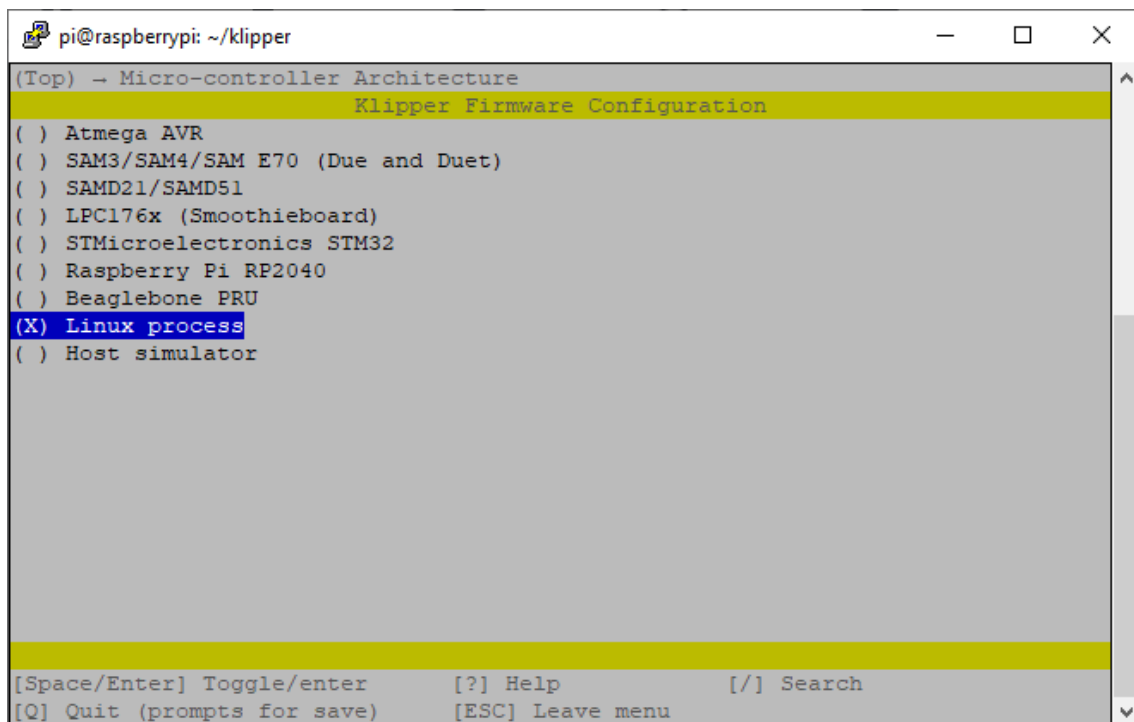
Este manual vai passar o passo a passo para que se possa fazer o acionamento ou leitura do estado dos pinos GPIOs da placa MKS Pi V1.0 e, futuramente, para a placa Orange Pi Zero 2.

A primeira coisa é que o sistema, no caso Linux, já esteja instalado juntamente com o Klipper, Moonraker e Mainsail/Fluidd.

Em seguida, vamos adicionar o processador da placa como uma segunda MCU, para isso, siga os passos abaixo (fonte: [https://www.klipper3d.org/RPi\\_microcontroller.html](https://www.klipper3d.org/RPi_microcontroller.html)):

1.  
`cd ~/klipper/`  
`sudo cp "./scripts/klipper-mcu-start.sh" /etc/init.d/klipper_mcu`  
`sudo update-rc.d klipper_mcu defaults`

2.  
`cd ~/klipper/`  
`make menuconfig`  
Nesse menu, selecione “Micro-controller Architecture (Linux process)”, use a barra de espaços para isso, depois digite Q para salvar e sair:



3.  
`sudo service klipper stop`  
`make flash`  
`sudo service klipper start`

Vamos adicionar a linha abaixo no arquivo printer.cfg:

```
[mcu host]
serial: /tmp/klipper_host_mcu
```

Projeto AGraber, por Antonio C. Guardia

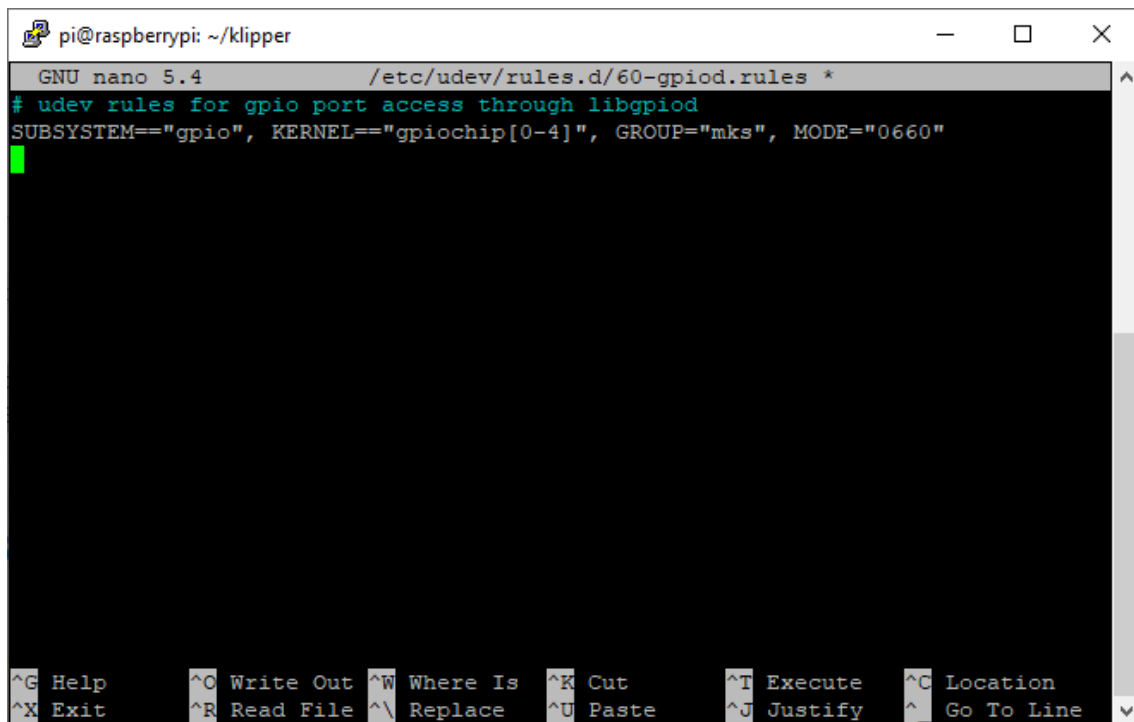
Em seguida, vamos criar um arquivo de configuração (regras) para dar acesso ao grupo de usuário, no caso da MKS PI é “mks” e no caso do Orange Pi Zero 2 é “orangepi”. Para isso digite no prompt:

```
sudo nano /etc/udev/rules.d/60-gpiod.rules
```

Copie e cole essas linhas no editor de texto que abriu:

```
# udev rules for gpio port access through libgpiod
SUBSYSTEM=="gpio", KERNEL=="gpiochip[0-4]", GROUP="mks", MODE="0660"
```

Pressione CTRL+X e, em seguida, Y para salvar e fechar o arquivo criado:

A screenshot of a terminal window titled 'pi@raspberrypi: ~/klipper'. The terminal shows the GNU nano 5.4 text editor editing the file '/etc/udev/rules.d/60-gpiod.rules'. The editor contains the following text: '# udev rules for gpio port access through libgpiod' and 'SUBSYSTEM=="gpio", KERNEL=="gpiochip[0-4]", GROUP="mks", MODE="0660"'. A green cursor is visible on the line following the rule. The bottom of the terminal shows the nano editor's command palette with options like Help, Write Out, Where Is, Cut, Execute, Location, Exit, Read File, Replace, Paste, Justify, and Go To Line.

Reinicie o sistema com:

```
sudo reboot
```

Após o reboot, instale o utilitário GpioID pelo comando:

```
sudo apt install gpiod
```

Sua execução se dá pelo comando:

```
sudo gpioinfo gpiochip
```

Esse utilitário vai te ajudar a encontrar os pinos configurados tanto no Klipper, quanto no Moonraker. Em seguida instale o WiringOP:

Projeto AGrabber, por Antonio C. Guardia

```
git clone https://github.com/orangepi-xunlong/wiringOP.git
cd wiringOP
./build clean
./build
```

Sua execução se dá pelo comando:

```
gpio readall
```

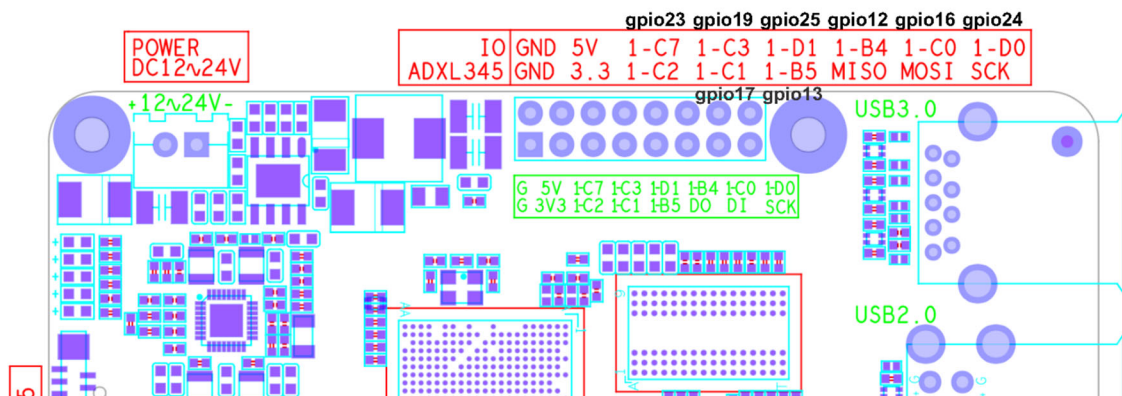
No Klipper, abra o arquivo moonraker.conf e adicione essas linhas:  
ATENÇÃO, NO ITEM “[file\_manager]” adicione apenas a linha em verde.

```
[file_manager]
queue_gcode_uploads: True
config_path: /home/pi/klipper_config
log_path: /home/pi/klipper_logs
```

```
[job_queue]
load_on_startup: True
```

```
[power AGrabber]
type: gpio
pin: gpiochip1/gpio23
off_when_shutdown: True
initial_state: off
on_when_job_queued: True
restart_klipper_when_powered: True
restart_delay: 5
```

A linha “pin: gpiochip1/gpio23” chama o pino 23, abaixo estão a relação dos pinos da placa MKS PI V1.0:  
Talvez seja necessário adicionar um “!” antes do “gpiochip1” para inverter o estado inicial do relé, ficando, portanto, dessa forma: “pin: !gpiochip1/gpio23”



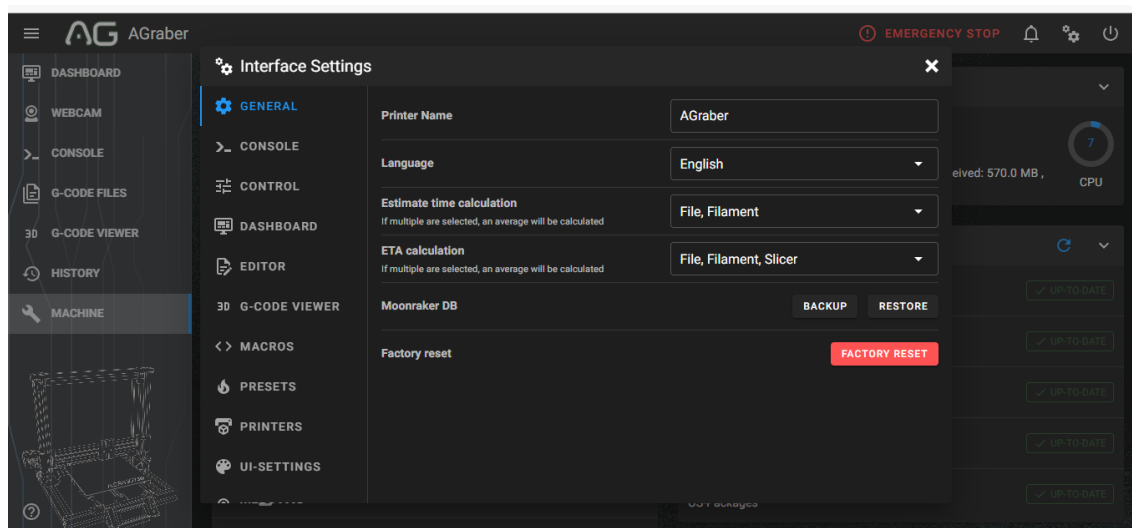
Para o Orange Pi Zero 2, o princípio é o mesmo, basta utilizar o esquema de GPIO da placa, que pode ser o 229, por exemplo, então ficaria “pin: gpiochip0/gpio229”:

```
pi@orangezero2:~$ gpio readall
```

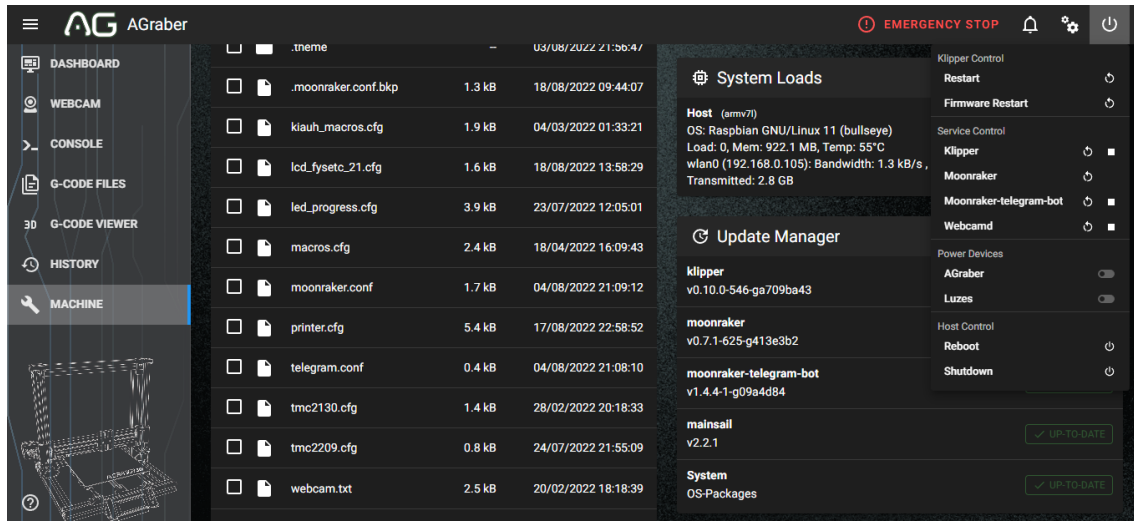
Zero 2											
GPIO	wPi	Name	Mode	V	Physical	V	Mode	Name	wPi	GPIO	
		3.3V			1	2		5V			
229	0	SDA.3	OUT	0	3	4		5V			
228	1	SCL.3	OFF	0	5	6		GND			
73	2	PC9	OFF	0	7	8	0	TXD.5	3	226	
		GND			9	10	0	RXD.5	4	227	
70	5	PC6	ALT5	0	11	12	0	PC11	6	75	
69	7	PC5	ALT5	0	13	14		GND			
72	8	PC8	OFF	0	15	16	0	PC15	9	79	
		3.3V			17	18	0	PC14	10	78	
231	11	MOSI.1	ALT4	0	19	20		GND			
232	12	MISO.1	ALT4	0	21	22	0	PC7	13	71	
230	14	SCLK.1	ALT4	0	23	24	0	CE.1	15	233	
		GND			25	26	0	PC10	16	74	
65	17	PC1	OFF	0	27	28					
272	18	PI16	ALT2	0	29	30					
262	19	PI6	OFF	0	31	32					
234	20	PH10	ALT3	0	33	34					

```
pi@orangezero2:~$
```

Lembrando que o nome que estiver na linha “[power AGraber]” deve ter o mesmo nome que estiver no menu “Interface Settings” -> “Printer Name”:



Com isso, no menu power do Mainsail ou Fluidd, irá aparecer um botão para ligar e desligar o relé, no submenu “Power Device”, e com isso, ligar e desligar sua impressora.



**Um agradecimento, muito querido, ao amigo Will que ajudou no descobrimento de como fazer o acionamento pelo GPIO.**

Agora vamos programar o Klipper pra desligar a impressora um tempo depois de terminar a impressão. Eu utilizo um arquivo chamado “macros.cfg” onde salvo todas as macros que uso na impressora. Caso você não faça uso de um arquivo separado, pode salvar junto com as outras macros no printer.cfg. Lembre-se do nome da impressora que foi falado lá em cima? Pois bem, aqui também precisa ser o mesmo nome que estiver na linha “[power AGraber]”:

```
[gcode_macro POWER_ON_PRINTER]
gcode:
    {action_call_remote_method("set_device_power",
        device="AGraber",
        state="on")}
#####

[gcode_macro POWER_OFF_PRINTER]
gcode:
    {action_call_remote_method("set_device_power",
        device="AGraber",
        state="off")}
#####

[idle_timeout]
gcode:
    M84
    TURN_OFF_HEATERS
    M117 Turning Off in 6 minutes
    UPDATE_DELAYED_GCODE ID=DELAYED_PRINTER_OFF DURATION=180
#####

[delayed_gcode DELAYED_PRINTER_OFF]
initial_duration: 0.
gcode:
    {% if printer.idle_timeout.state == "Idle" %}
        POWER_OFF_PRINTER
```

Projeto AGraber, por Antonio C. Guardia

{% endif %}

#####