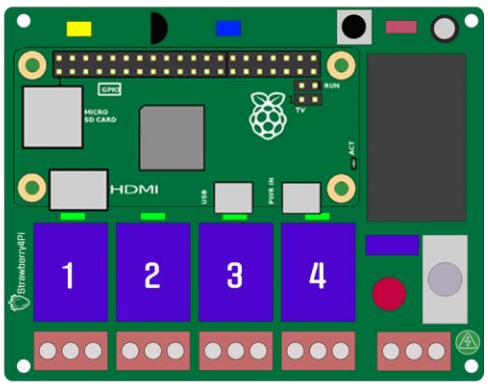
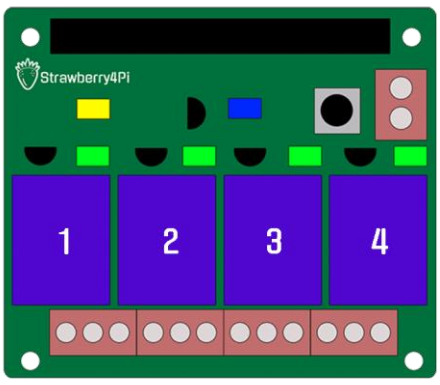
Strawberry4Pi
for developers

Index



Strawbery4Pi Zero

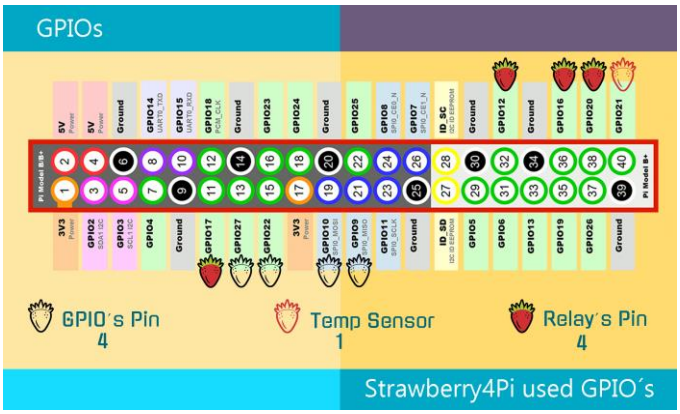


Strawbery4Pi Shield

Relay 1 = GPIO12 - Relay 2 = GPIO16 - Relay 3 = GPIO20 - Relay 4 = GPIO17

Strawberry4Pi Raspbian documentation

Strawberry4Pi OS is an image installed of Raspbian for Raspberry Pi with everything you need to start your project quickly. Download the Smartphone App and create services joining gpios from different Raspberry Pis. You can also make them work together at once, as if they were only one device. S4Pi uses this GPIOs configuration:



Main path for S4Pi files:

This folder contains the backup files to recover the device in case it would be necessary

`/home/pi/Documents/BK_OrionSky`

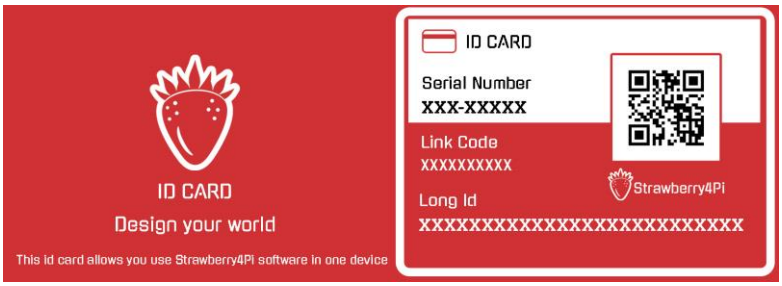
The Main folder on where we will be working

`/home/pi/Documents/OrionSky`

IOTClient.exe

This is the manager of every communication between de OS and S4Pi app. You must not change anything in this file.

Config



ID Card: This numerical key code will enable the connection between the cloud and S4Pi App. During the first configuration you can use the ID card either manually or scanning the QR code on Adhoc mode.

`GUID=XXXXXXXX-XXXXX-XXXX-XXXX-XXXXXXXXXXXX;SERIAL=BOX-XXXXX;LINKCODE=XXXXXXXXXX;SERVERNAME=www.orioncloud.es:5001;`

ExecutedTimer

This file is the trigger that sends the current status of the gpios to the S4Pi App. Usually the value is “N”. If you want to force the update status change this value to “DN”.

Example:

`sudo echo ‘DN’ > /home/pi/Documents/OrionSky/ExecutedTimer`

pinStatus

Last status of gpios and thermostat info. Also you can find the current Local_IP

`%Pin1=0;Pin2=0;Pin3=0;Pin4=0;Pin5=0;Pin6=0;Pin7=0;Pin8=0;ecoMode=0;thStatus=0;thId=86;setTemp=28;ip=192.168.1.33;%`

Boot Script

S4Pi OS has a script called StartScript.sh located in /etc/init.d/ which is used in the boot to start and configure everything that the software needs to operate: gpio configuration, button setup, wifi led…

Here you can call your script to boot it when Raspbian is booting.

```
sudo nano /etc/init.d/StartScript.sh
```

add it between this tags:

```
#your custom scripts

sudo python /home/pi/Documents/yourFolder/yourCustomScript.py &

#end your custom scripts
```

Save file and reboot.

Custom code

If you like to create your custom codes to improve the out of the box features of S4Pi, you can also do it while getting the most out of all the potential of raspberry pi and Strawberry4Pi together. For example, create your Python script to check if gpioX value = 1 or 0 then execute a task. Then, when you switch this gpio from the App you will be able to start your custom task.

This new script could be executed during boot. Add it to Strawberry4Pi Start Script.

Thermostat

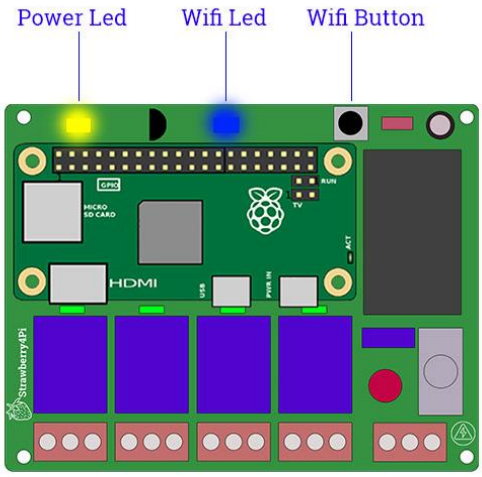
This is the file that reads the temperature from S4Pi Shield sensor. If you think you need to change something like, correct the temperature to obtain a more accurate measure, you can do it here:

```
/home/pi/Documents/OrionSky/Temperature/temp.py
```

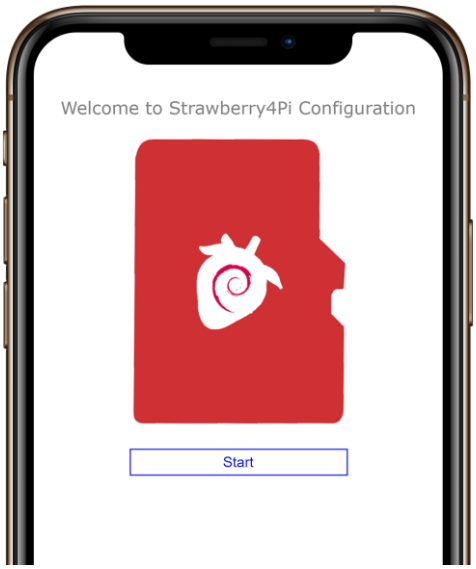
The final value is written in this file in ºC:

```
/home/pi/Documents/OrionSky/Temperature/TempCurrent
```

AD HOC MODE

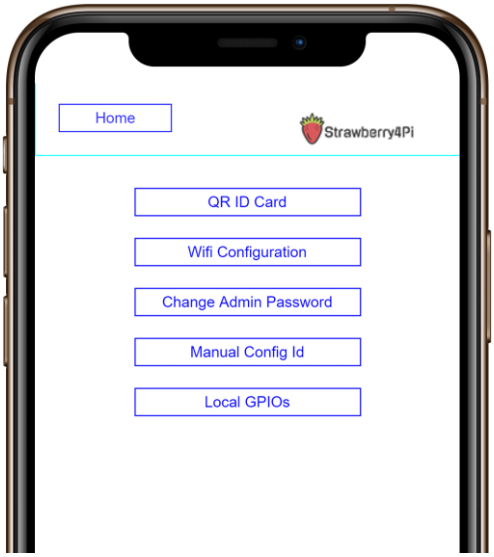


1. Burn your micro SD memory card with **Prepared Raspbian** and introduce it on your Raspberry Pi Zero W
    1. Look at this manual.
2. Plug Raspberry Pi Zero W on your **Strawberry4Pi**
3. Switch it on and you will see the power LED yellow turning on
4. Wait until the blue LED starts to blink
5. Push and hold the WiFi button **during 6 seconds** and release it
6. You will see that the blue LED starts to blink faster
7. Go to your SmartPhone WiFi configuration and **connect** to the new WiFi called **"strawberry4pi"**, password **"strawberry"**
8. Now open your web navigator, for example chrome, firefox, safari... and go: **http://straw.berry**



Click on the Start button. If your browser shows “your connection is not private” then you should click on “advance configuration” and “Go to site no safe”

9. You will see a popup windows asking for your user and password. **User: admin Pass: admin**



10. Go to QR ID Card menu and complete the **info about your serial number** from your Strawberry4Pi. If you have any problems with your camera you can complete this info manually in **"Manual Config ID"**
11. Go to WiFi configuration and select your **WiFi**, introduce your **password** and click connect.
12. You will see the **blue LED light remains fixed.** That means we are connected to Strawberry App.
13. Ensure that your smartphone has internet connection and go to **Strawberry4pi App.**

User Password

To ensure the security in your device you must change the default password admin.

SSH Connection

The default user and password is:

User: S4Pi

Password: strawberry

Root user is disable by default.

Extend micro sd card partition

If you want to use the whole microsd space you can extend your partition easily.

Execute this command:

```
sudo raspi-config --expand-rootfs
```

after

```
reboot
```

To check if you are using the whole space of your microsd card you can execute:

```
df -h
```

```
root@Strawberry4Pi:/home# df -h
S.ficheros     Tamaño Usados  Disp Uso% Montado en
/dev/root         30G   1,9G   27G   7% /
devtmpfs         456M      0  456M   0% /dev
tmpfs            462M      0  462M   0% /dev/shm
tmpfs            462M   6,5M  456M   2% /run
tmpfs            5,0M   4,0K  5,0M   1% /run/lock
tmpfs            462M      0  462M   0% /sys/fs/cgroup
/dev/mmcblk0p1    41M    21M   21M  52% /boot
tmpfs             93M   4,0K   93M   1% /run/user/1000
tmpfs             93M      0   93M   0% /run/user/0
```

LOCAL GPIOs

Strawberry4Pi has a local GPIOs control **synchronized with the App** which allows you to switch your GPIOs through your LAN.

Control your GPIOs from this local web located in Strawberry4Pi OS and keep the GPIOs status synchronized with the App Service Information. Of course that we have thought in your maker soul and S4Pi has enabled for you a way to control GPIOs status programmatically. (esto creo q esta bien, pero no pillo bien el sentido, para q se quede programado no? Lo del alma tampoco lo he pillado)

`https://LOCAL_IP:448/gpio_demo.php?pin=GPIO_NUMBER&value=VALUE&api=LINKCODE`

Example:

`Turn On gpio12`

`https://192.168.1.33:448/gpio_demo.php?pin=12&value=1&api=12345`

`Turn Off gpio12`

`https://192.168.1.33:448/gpio_demo.php?pin=12&value=0&api=12345`