

Tutorial para conectar la Raspberry PI a Internet con el Shield 3G/4G/4.5G/4G+/LTE/GPS/Glonass

Paso 1: preparar una tarjeta de memoria micro SD con el sistema operativo. Para esto se debe entrar al sitio web oficial de Raspberry PI y descargar la última versión de Raspbian.

<https://www.raspberrypi.org/downloads/>

Luego, la forma más fácil de cargar el sistema operativo en la memoria micro SD, es mediante el software Etcher.

<https://www.balena.io/etcher/>

Este procedimiento es tan sencillo e intuitivo, que consideramos que no requiere mayor explicación.

Paso 2: Insertar la tarjeta de memoria en la Raspberry, instalar el Shield 4G y conectar los periféricos que vayamos a utilizar. Lo más fácil es contar con un mouse y un teclado y un televisor o monitor que acepte conexión HDMI. También es necesario instalar un pigtail y una antena para que el Shield pueda conectarse a la red celular.

El Pigtail podría ser este:

<https://altronics.cl/cable-pigtail-sma-hembra-ipex-ufl-15cm>

y la antena podría ser esta u otra que sirva para las frecuencias que usamos en Chile:

<https://altronics.cl/antena-gsm-quadband-sma-5dbi>

El módem 4G tiene tres conectores para antena:

- 1.- MAIN, que es la conexión principal a la red celular.
- 2.- GNSS, que es para conectar la antena GPS/GLONASS
- 3.- DIVERSITY, que es para conectar una segunda antena celular.

Esta tercera conexión, corresponde a una segunda antena celular opcional, que se utiliza para mejorar la señal cuando el módem está en movimiento. Entonces, nuestra recomendación es que si tu aplicación es para un sistema en movimiento, utilices dos antenas celulares.

El Pigtail indicado anteriormente sirve para las tres conexiones de antena.

También es necesario insertar la tarjeta micro SIM en el Shield, antes de energizar.



El Shield 4G incluye un cable USB corto, que está pensado para establecer comunicación con la Raspberry y también para la alimentación del Shield.



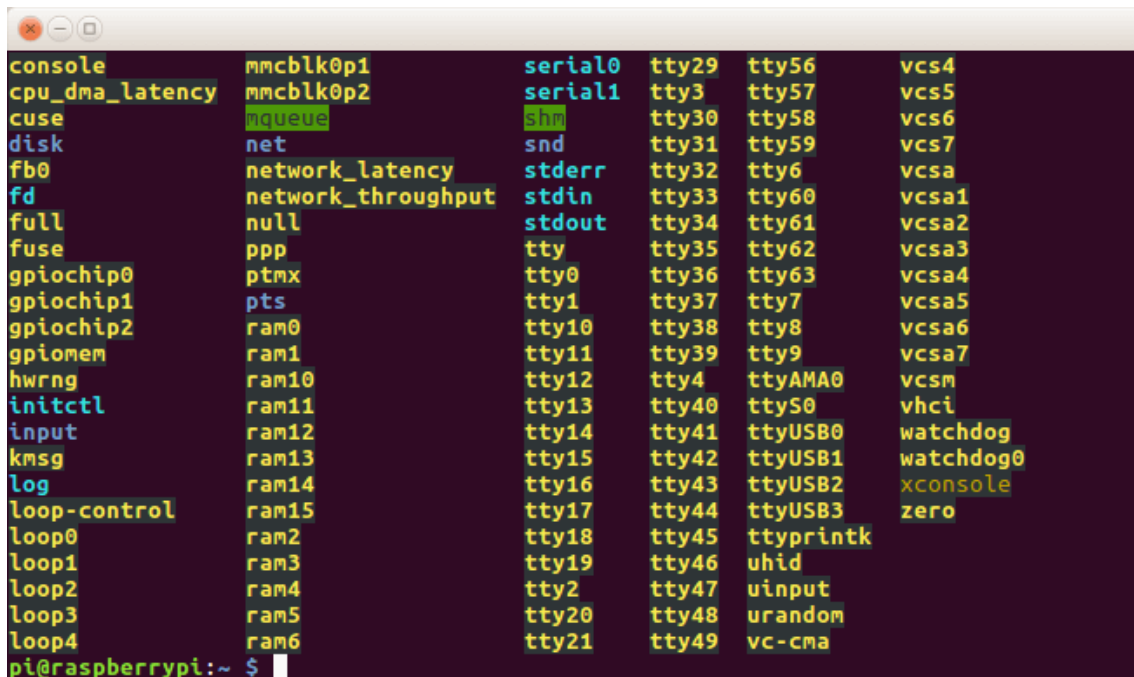
En realidad el Shield también se puede utilizar sin conexión USB, pero en ese caso tomará la alimentación desde los pines GPIO de la Raspberry y la comunicación se realizará a través de puerto serial (UART) TTL. Esta otra forma tiene como inconveniente que la velocidad de conexión a Internet (ancho de banda) será menor.

El fabricante advierte que en caso de establecer la comunicación vía UART, la máxima tasa de transferencia de datos llegará a los 900 Kbit/s.

Paso 3: Conectar la energía y dejar que el sistema operativo se inicie y se actualice. Para esto es necesario que la Raspberry esté conectada a Internet, normalmente mediante Wifi o cable de red Ethernet. La conexión a Internet será necesaria mediante todo este procedimiento, hasta que tengamos todo listo y configurado.

Paso 4: Verificar que el sistema operativo reconoció el módem 4G. Para esto, debemos abrir un terminal y enviar el siguiente comando:

ls /dev



```
console      mmcblk0p1    serial0      tty29        tty56        vcs4
cpu_dma_latency mmcblk0p2    serial1      tty3         tty57        vcs5
cuse         mqueue       shm          tty30        tty58        vcs6
disk         net          snd          tty31        tty59        vcs7
fb0          network_latency stderr        tty32        tty6         vcsa
fd           network_throughput stdin         tty33        tty60        vcsa1
full        null        stdout       tty34        tty61        vcsa2
fuse        ppp         tty          tty35        tty62        vcsa3
gpiochip0   ptmx        tty0         tty36        tty63        vcsa4
gpiochip1   pts         tty1         tty37        tty7         vcsa5
gpiochip2   ram0        tty10        tty38        tty8         vcsa6
gpiomen     ram1        tty11        tty39        tty9         vcsa7
hwrng       ram10       tty12        tty4         ttyAMA0      vcsn
initctl     ram11       tty13        tty40        ttyS0        vhci
input       ram12       tty14        tty41        ttyUSB0      watchdog
kmsg        ram13       tty15        tty42        ttyUSB1      watchdog0
log         ram14       tty16        tty43        ttyUSB2      xconsole
loop-control ram15       tty17        tty44        ttyUSB3      zero
loop0       ram2        tty18        tty45        ttyprintk
loop1       ram3        tty19        tty46        uhid
loop2       ram4        tty2         tty47        uinput
loop3       ram5        tty20        tty48        urandom
loop4       ram6        tty21        tty49        vc-cma
pi@raspberrypi:~ $
```

Si todo está bien, deberían aparecer los puertos USB:

ttyUSB0

ttyUSB1

ttyUSB2

ttyUSB3

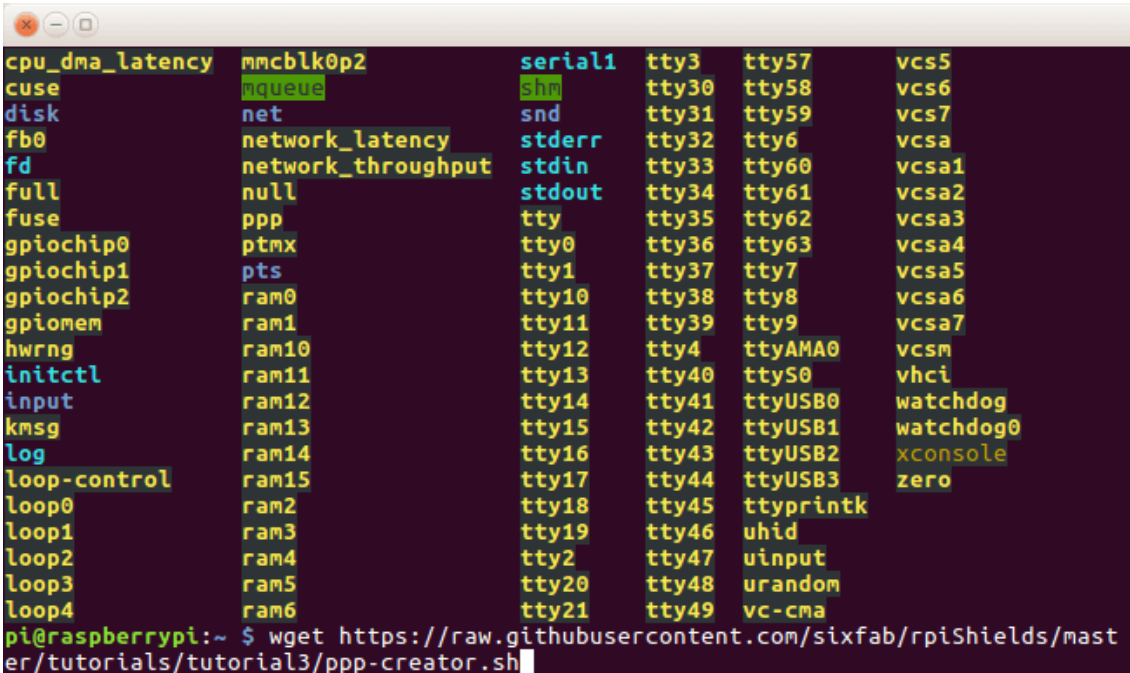
Normalmente el puerto ttyUSB3 será el que utilizaremos para establecer la conexión a Internet. El puerto ttyUSB2 lo utilizaremos para enviar comandos AT al módem y el puerto ttyUSB1 lo usaremos para recibir los datos del GPS, en caso que lo necesitemos. Para usar el GPS será necesario instalar un Pigtail y una antena como estos:

<https://altronics.cl/cable-pigtail-sma-hembra-ipex-ufl-15cm>

<https://altronics.cl/ant-gps-28db>

Paso 5: descargar el script que creará la conexión PPP a Internet, mediante el siguiente comando:

wget <https://raw.githubusercontent.com/sixfab/rpiShields/master/tutorials/tutorial3/ppp-creator.sh>



```
cpu_dma_latency mmcblk0p2 serial1 tty3 tty57 vcs5
cuse wqueue shm tty30 tty58 vcs6
disk net snd tty31 tty59 vcs7
fb0 network_latency stderr tty32 tty6 vcsa
fd network_throughput stdin tty33 tty60 vcsa1
full null stdout tty34 tty61 vcsa2
fuse ppp tty tty35 tty62 vcsa3
gpiochip0 ptmx tty0 tty36 tty63 vcsa4
gpiochip1 pts tty1 tty37 tty7 vcsa5
gpiochip2 ram0 tty10 tty38 tty8 vcsa6
gpiomem ram1 tty11 tty39 tty9 vcsa7
hwrng ram10 tty12 tty4 ttyAMA0 vcsn
initctl ram11 tty13 tty40 ttyS0 vhci
input ram12 tty14 tty41 ttyUSB0 watchdog
kmsg ram13 tty15 tty42 ttyUSB1 watchdog0
log ram14 tty16 tty43 ttyUSB2 xconsole
loop-control ram15 tty17 tty44 ttyUSB3 zero
loop0 ram2 tty18 tty45 ttyprintk
loop1 ram3 tty19 tty46 uhid
loop2 ram4 tty2 tty47 uinput
loop3 ram5 tty20 tty48 urandom
loop4 ram6 tty21 tty49 vc-cma
pi@raspberrypi:~ $ wget https://raw.githubusercontent.com/sixfab/rpiShields/master/tutorials/tutorial3/ppp-creator.sh
```

Luego ejecutar los siguientes comandos:

```
chmod +x ./ppp-creator.sh
```

```
sudo ./ppp-creator.sh HOLOGRAM ttyUSB3
```

En este caso se debe cambiar la palabra “HOLOGRAM” por el nombre del APN que vayamos a utilizar.

Si además del nombre de APN el operador móvil requiere un nombre de usuario y contraseña (como en el caso de Entel), será necesario editar el archivo `/etc/ppp/peers/gprs` haciendo los siguientes cambios:

- Remover o comentar la línea “noauth”.
- Agregar las siguientes dos líneas:

```
user “XXXXXXXXXX”
```

```
password “ZZZZZZZZZ”
```

Reemplazar las X y las Z por el usuario y contraseña que correspondan.

Luego deberemos agregar la siguiente línea al final del archivo `/etc/ppp/ip-up`

```
sudo route add default ppp0
```

Paso 6: Luego de ejecutar los comandos indicados, ya estamos en condiciones de desconectar la conexión a Internet (Wifi o Ethernet) y podremos conectarnos a internet mediante el Shield ejecutando el siguiente comando:

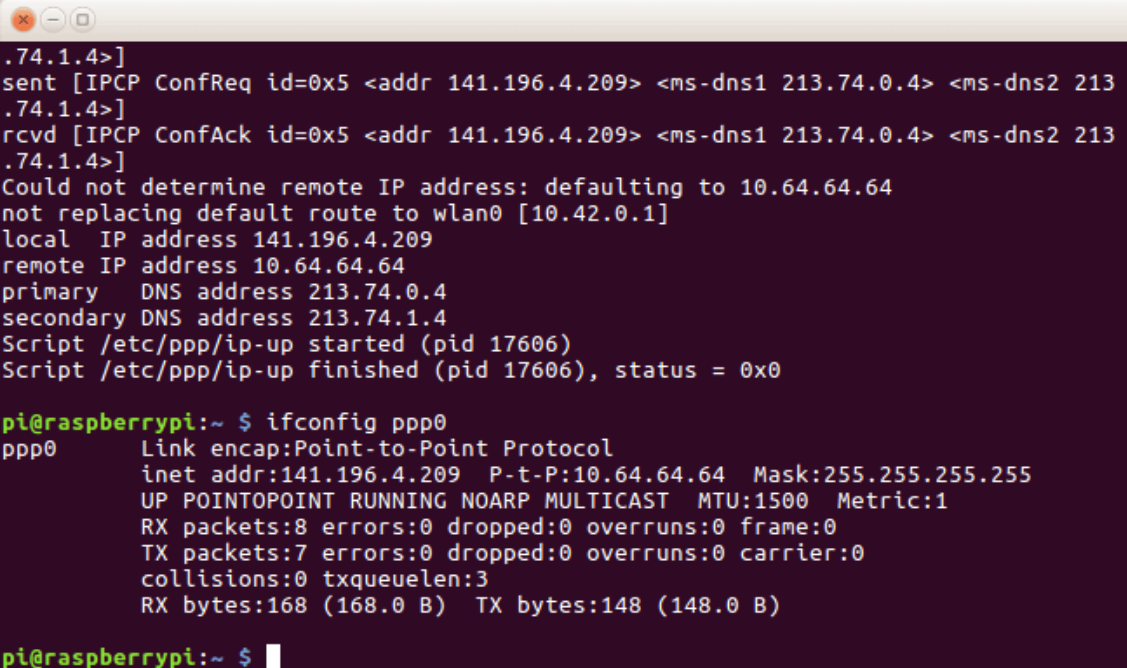
```
sudo pppd call gprs
```

Pero si queremos que el comando se ejecute en segundo plano (background) tendremos que usar este otro comando:

```
sudo pppd call gprs&;
```

Si todo salió bien, ya deberíamos tener conexión a Internet y podemos verificar que se recibió dirección IP mediante el siguiente comando:

```
ifconfig ppp0
```



```
.74.1.4>]
sent [IPCP ConfReq id=0x5 <addr 141.196.4.209> <ms-dns1 213.74.0.4> <ms-dns2 213
.74.1.4>]
rcvd [IPCP ConfAck id=0x5 <addr 141.196.4.209> <ms-dns1 213.74.0.4> <ms-dns2 213
.74.1.4>]
Could not determine remote IP address: defaulting to 10.64.64.64
not replacing default route to wlan0 [10.42.0.1]
local IP address 141.196.4.209
remote IP address 10.64.64.64
primary DNS address 213.74.0.4
secondary DNS address 213.74.1.4
Script /etc/ppp/ip-up started (pid 17606)
Script /etc/ppp/ip-up finished (pid 17606), status = 0x0

pi@raspberrypi:~ $ ifconfig ppp0
ppp0      Link encap:Point-to-Point Protocol
          inet addr:141.196.4.209 P-t-P:10.64.64.64  Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
          RX packets:8 errors:0 dropped:0 overruns:0 frame:0
          TX packets:7 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:3
          RX bytes:168 (168.0 B)  TX bytes:148 (148.0 B)

pi@raspberrypi:~ $
```

Pero si no logras establecer conexión y te aparece un error asociado a la autenticación de la conexión PPP, deberás eliminar o comentar la línea que dice "auth" en el archivo

```
/etc/ppp/options
```

Seguramente hay muchas cosas que se pueden mejorar en este tutorial, pero siguiendo estos pasos obtuvimos una conexión exitosa.

Si quieres aportar algo, por favor escríbenos usando el formulario de contacto que tenemos en la tienda virtual <https://altronics.cl>

<https://altronics.cl/index.php?route=information/contact>