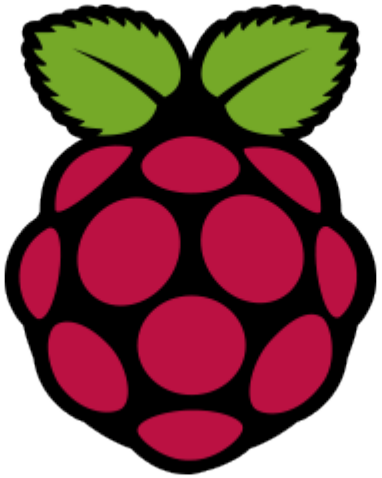
Comunicacions amb Raspberry Pi



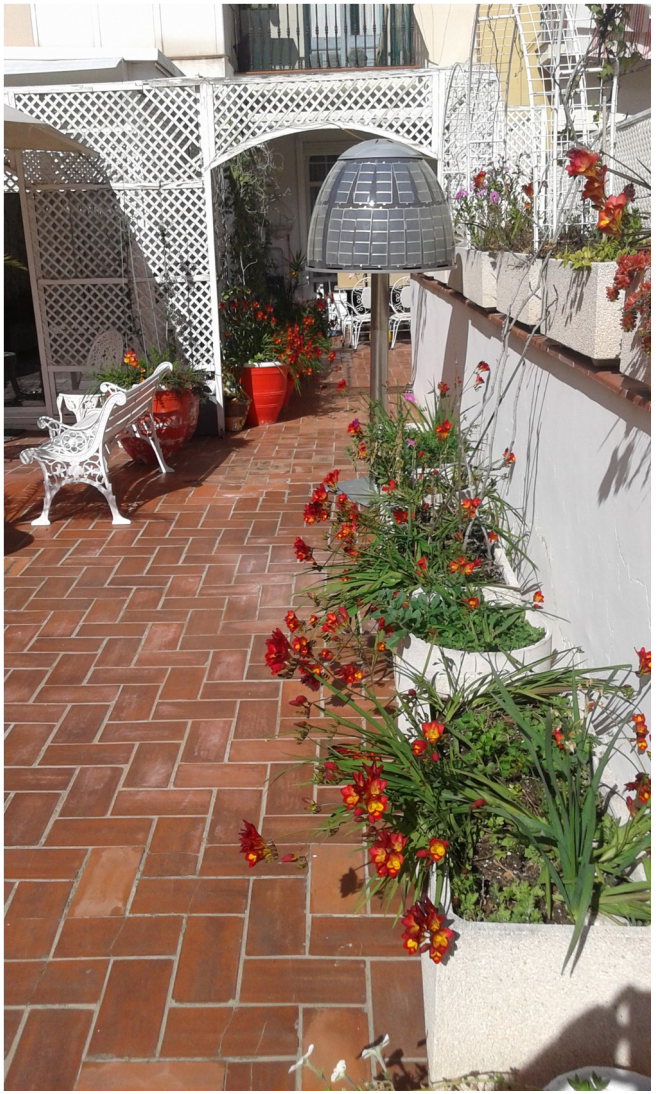
JESUÏTES El Clot
Escola del Clot

11 de març de 2017

Jordi Binefa i Martínez

Comunicacions amb Raspberry Pi

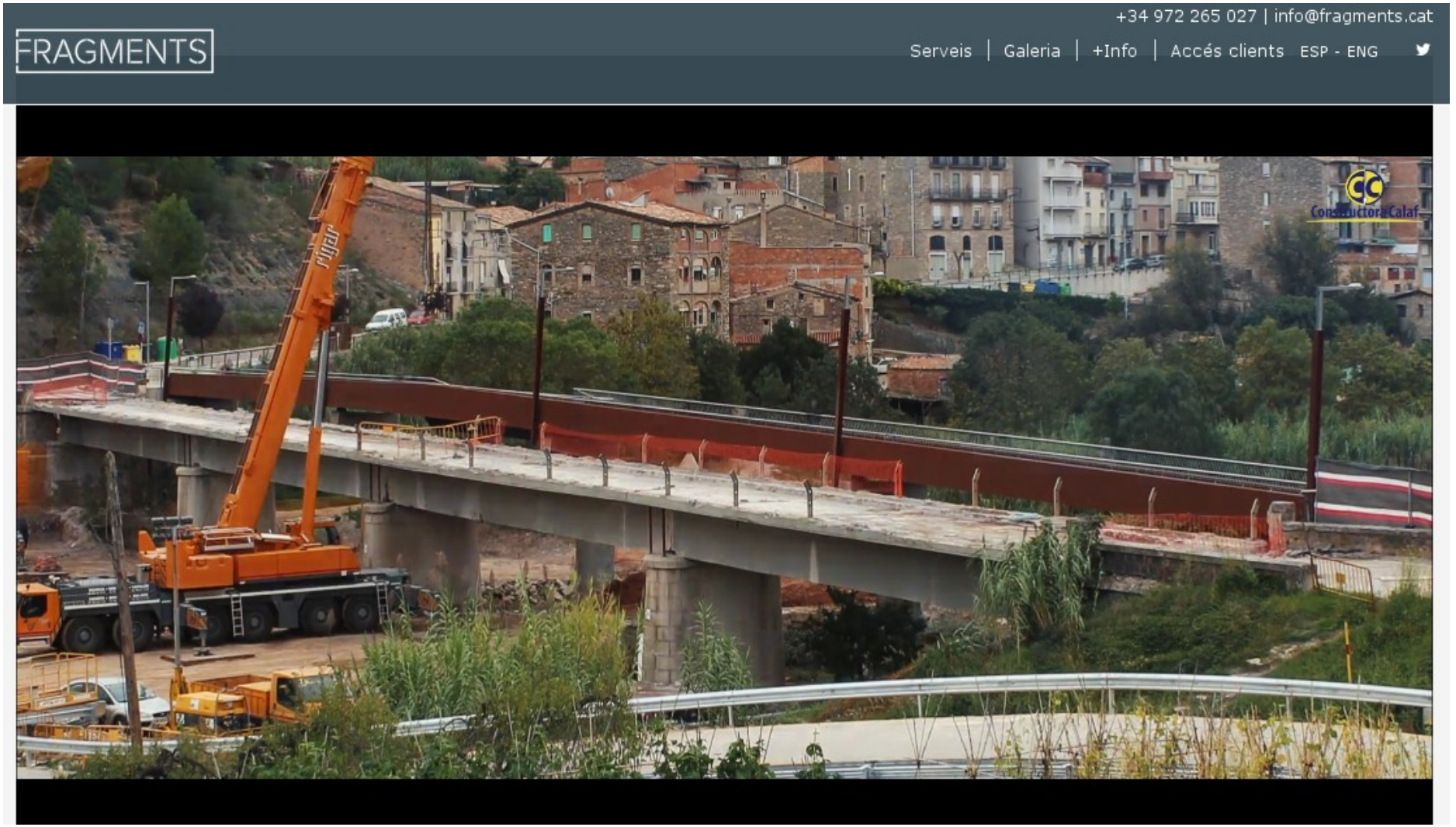
Exemples



http://siarq.com/

Comunicacions amb Raspberry Pi

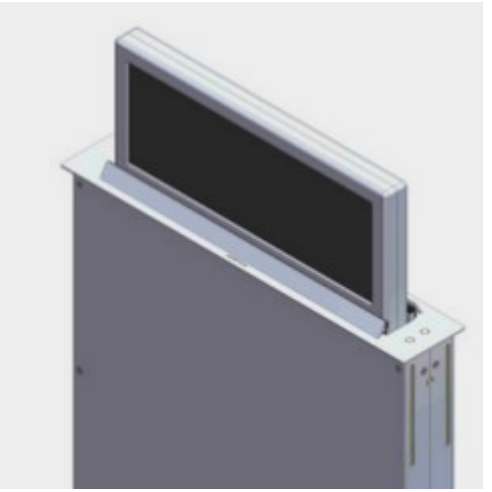
Exemples



http://www.fragments.cat/projectes-seguiment-obres/

Comunicacions amb Raspberry Pi

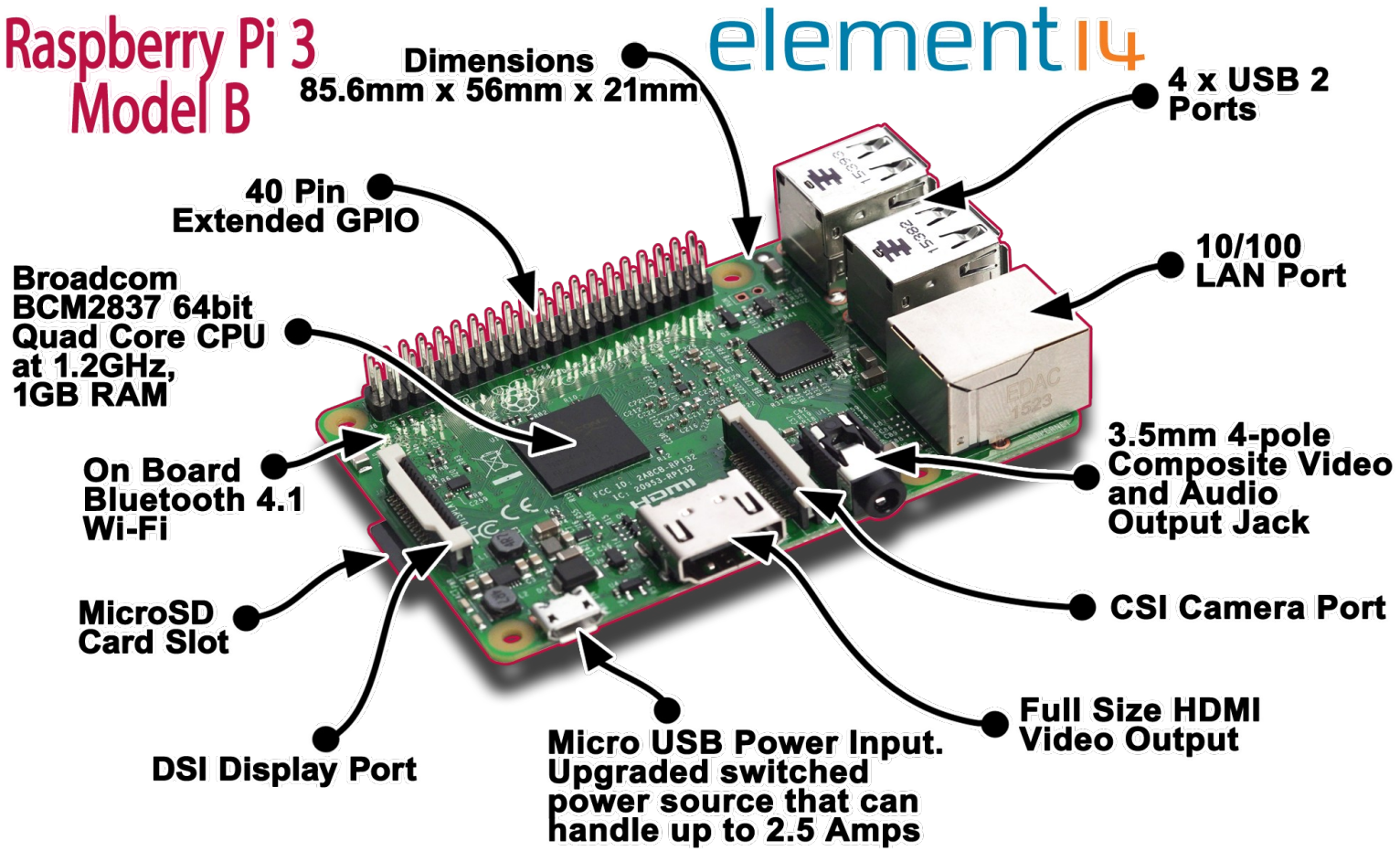
Exemples



http://www.arthurholm.com/

Comunicacions amb Raspberry Pi

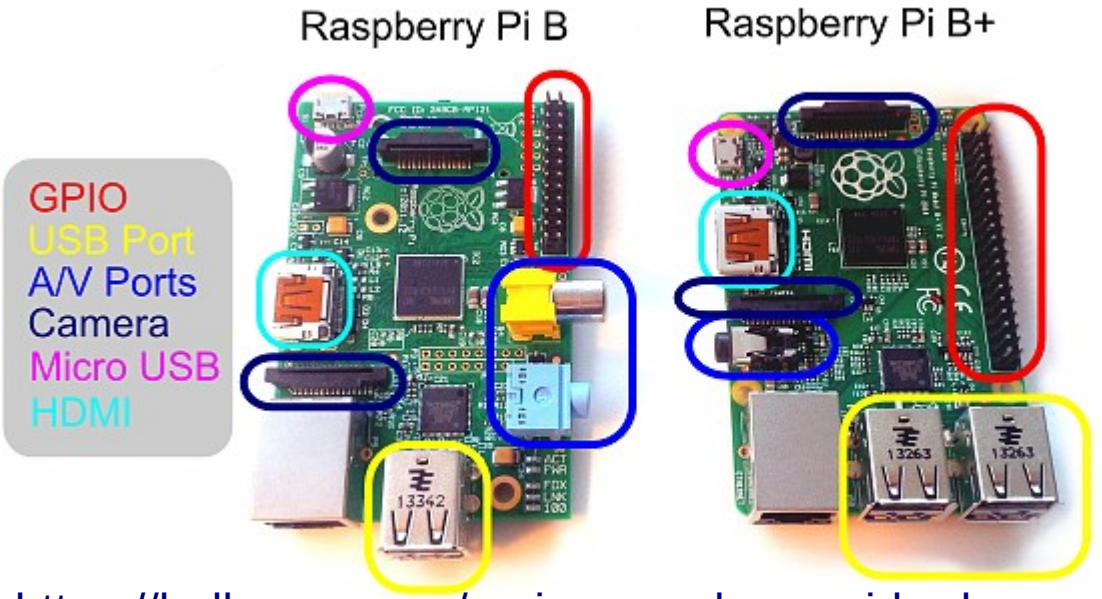
La Raspberry Pi



http://www.raspberrypi.org/

Comunicacions amb Raspberry Pi

La Raspberry Pi



https://helloacm.com/review-raspberry-pi-b-plus-model/

http://misapuntesde.com/post.php?id=431

Comunicacions amb Raspberry Pi

La Raspberry Pi

- Sortida de vídeo HDMI i vídeo compost
- Es poden fer servir connectors DVI o VGA amb un adaptor
- Les pantalles tàctils HDMI funcionen si emulen un ratolí USB
- Chalkboard Electronics té una pantalla tàctil compatible de 10 polsades amb una interfície d'HDMI a LVDS

Comunicacions amb Raspberry Pi

La Raspberry Pi

- Ports GPIO, sèrie, SPI, I2C i JTAG
- Ve amb un connector soldat (P1) de 26 potes i un altre de 8 potes (P5. Tan sols està a la segona revisió)
- Treballa internament a 3,3V. Si es vol fer compatible amb nivells TTL (5V) cal un convertidor de nivells
- MIPI CSI-2 (Camera Serial Interface) al connector S5
- DSI (Display Serial Interface) al connector S2 per a LCD (no hi ha drivers actualment)
- La segona revisió té alguns canvis menors respecte a l'original

Connector GPIO

GPIO Numbers

Raspberry Pi B
Rev 1 P1 GPIO Header

	Pin No.		
3.3V	1	2	5V
GPIO0	3	4	5V
GPIO1	5	6	GND
GPIO4	7	8	GPIO14
GND	9	10	GPIO15
GPIO17	11	12	GPIO18
GPIO21	13	14	GND
GPIO22	15	16	GPIO23
3.3V	17	18	GPIO24
GPIO10	19	20	GND
GPIO9	21	22	GPIO25
GPIO11	23	24	GPIO8
GND	25	26	GPIO7

Raspberry Pi A/B
Rev 2 P1 GPIO Header

	Pin No.		
3.3V	1	2	5V
GPIO2	3	4	5V
GPIO3	5	6	GND
GPIO4	7	8	GPIO14
GND	9	10	GPIO15
GPIO17	11	12	GPIO18
GPIO27	13	14	GND
GPIO22	15	16	GPIO23
3.3V	17	18	GPIO24
GPIO10	19	20	GND
GPIO9	21	22	GPIO25
GPIO11	23	24	GPIO8
GND	25	26	GPIO7

Raspberry Pi B+
B+ J8 GPIO Header

	Pin No.		
3.3V	1	2	5V
GPIO2	3	4	5V
GPIO3	5	6	GND
GPIO4	7	8	GPIO14
GND	9	10	GPIO15
GPIO17	11	12	GPIO18
GPIO27	13	14	GND
GPIO22	15	16	GPIO23
3.3V	17	18	GPIO24
GPIO10	19	20	GND
GPIO9	21	22	GPIO25
GPIO11	23	24	GPIO8
GND	25	26	GPIO7
DNC	27	28	DNC
GPIO5	29	30	GND
GPIO6	31	32	GPIO12
GPIO13	33	34	GND
GPIO19	35	36	GPIO16
GPIO26	37	38	GPIO20
GND	39	40	GPIO21

Key

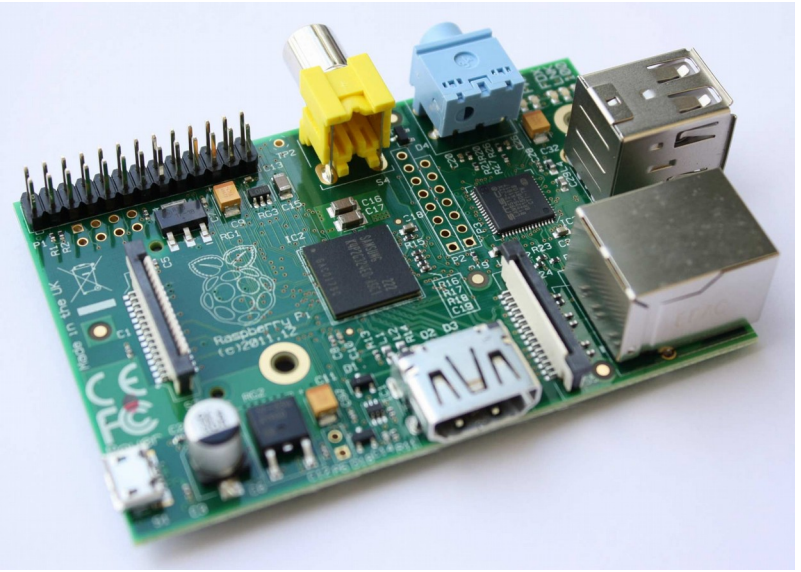
Power +	UART
GND	SPI
I²C	GPIO

Font : http://raspi.tv/2014/rpi-gpio-quick-reference-updated-for-raspberry-pi-b

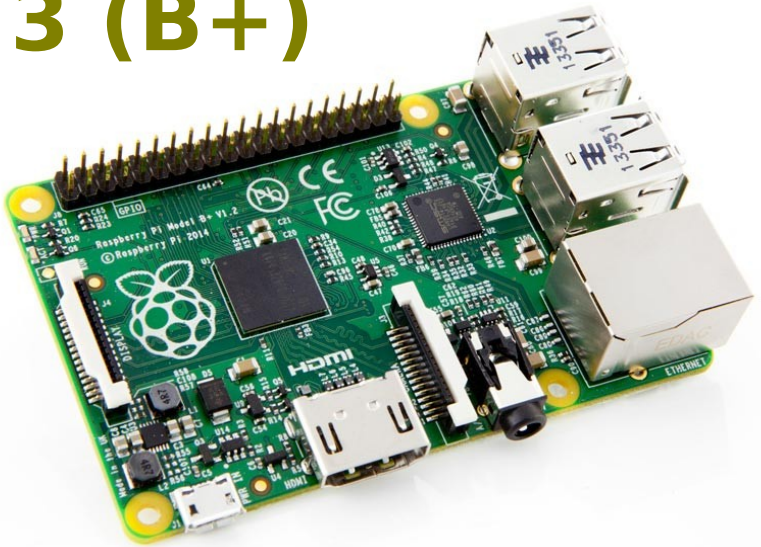
Raspberry Pi Model B

Rev 1

Rev 2

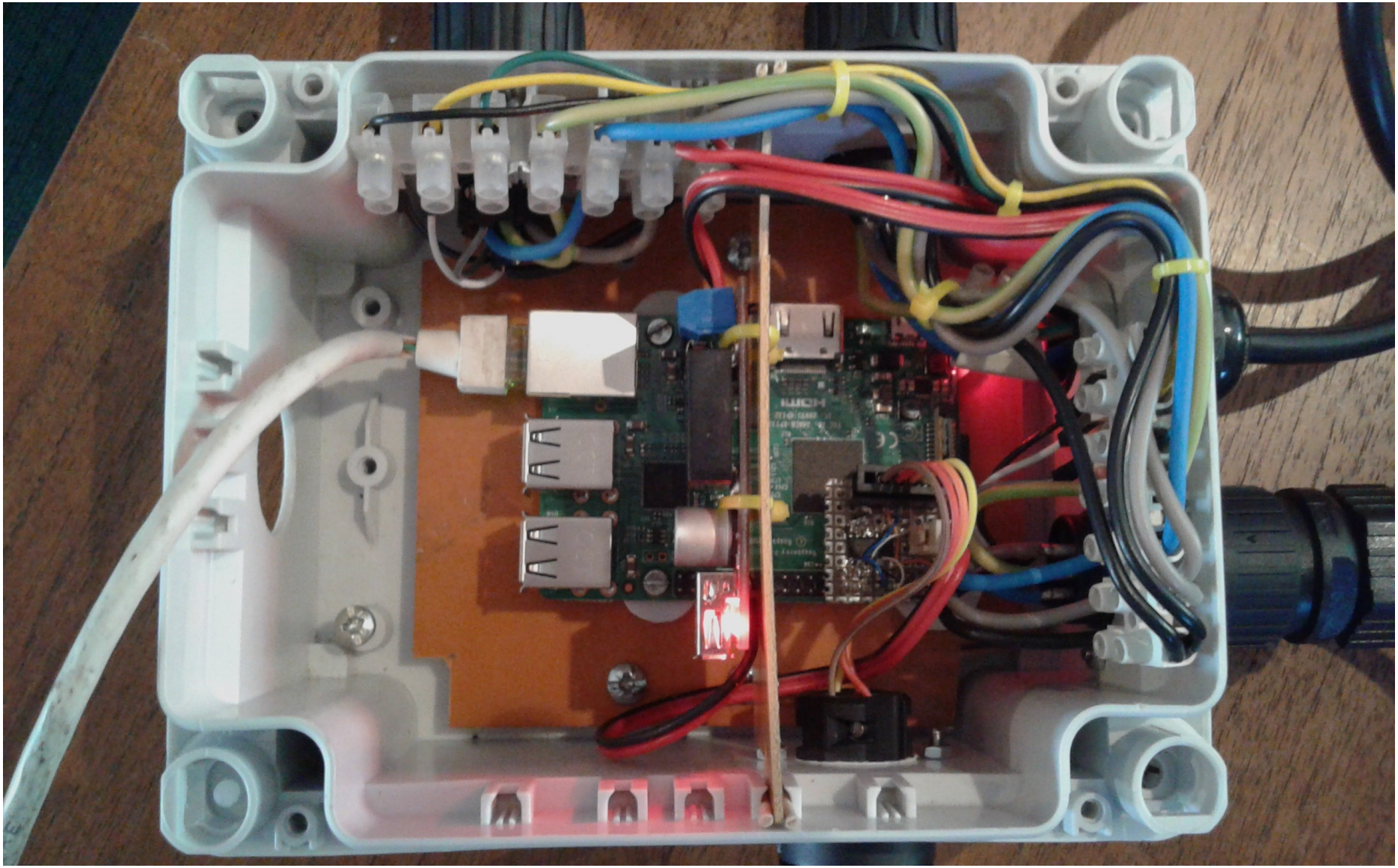


Rev 3 (B+)



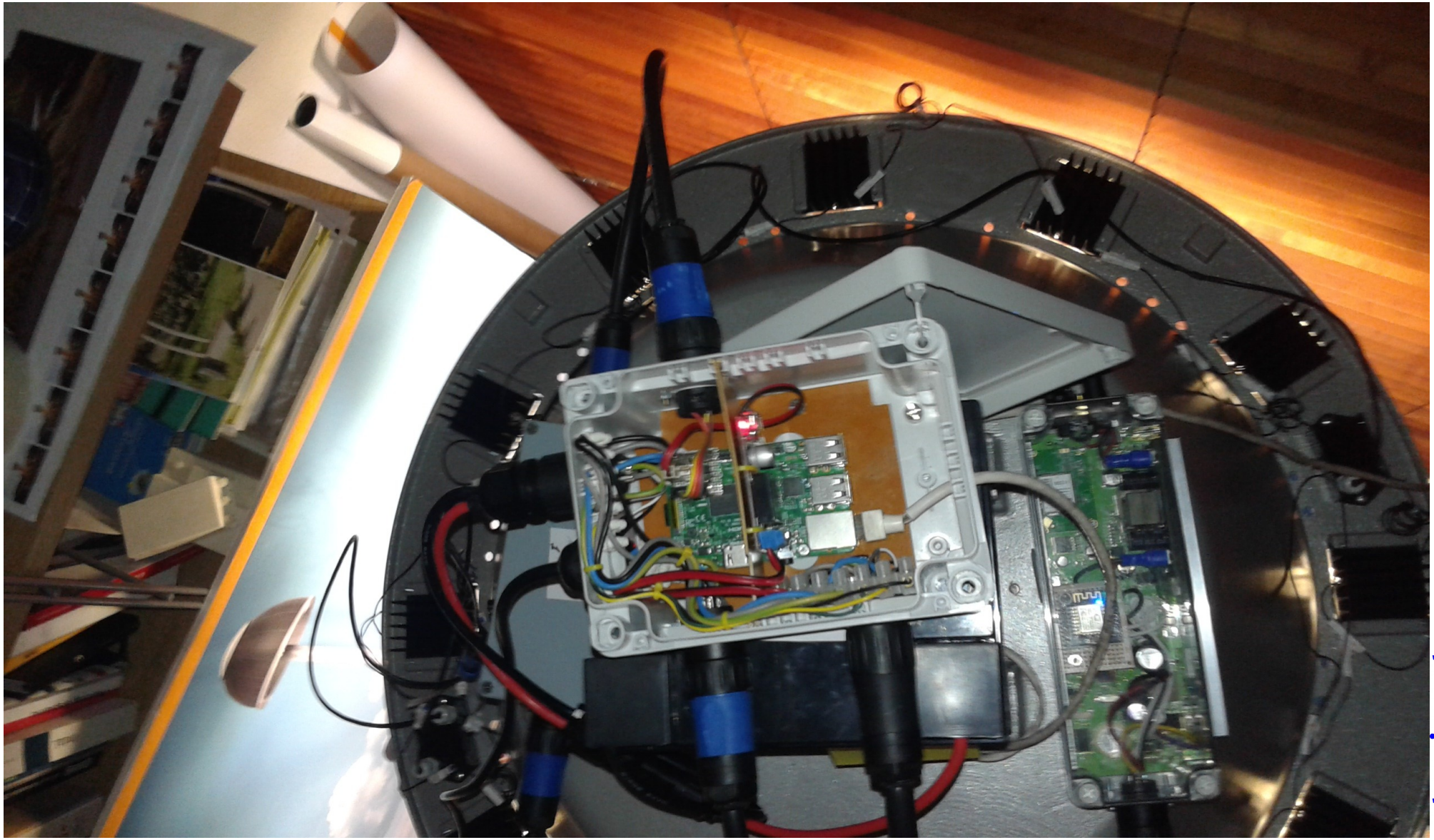
Comunicacions amb Raspberry Pi

Connexió amb altre maquinari



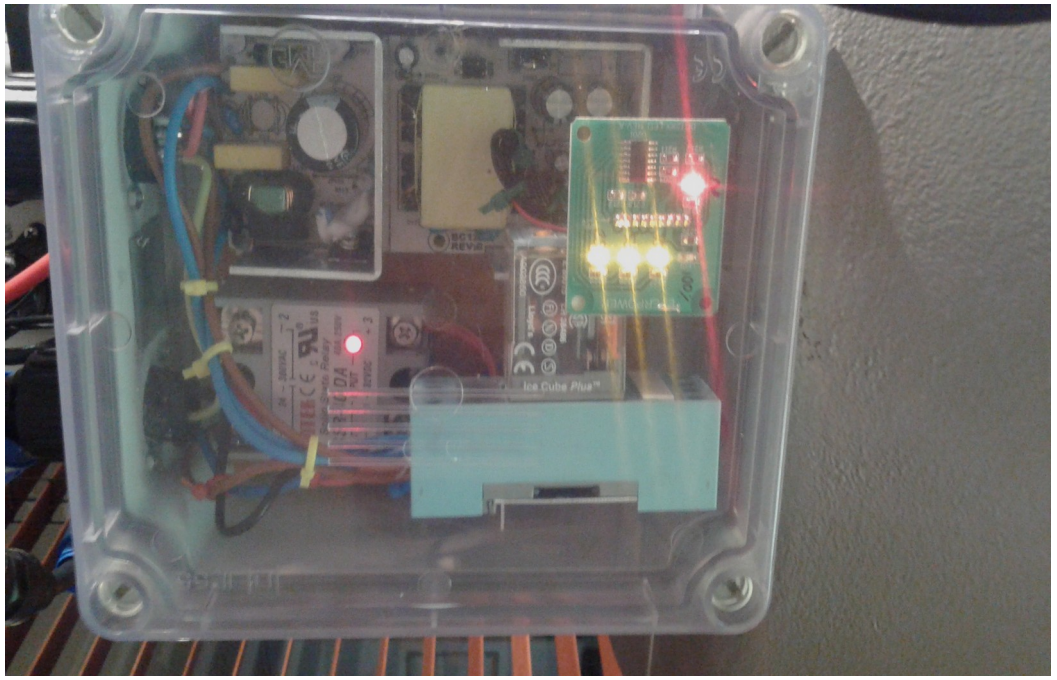
Comunicacions amb Raspberry Pi

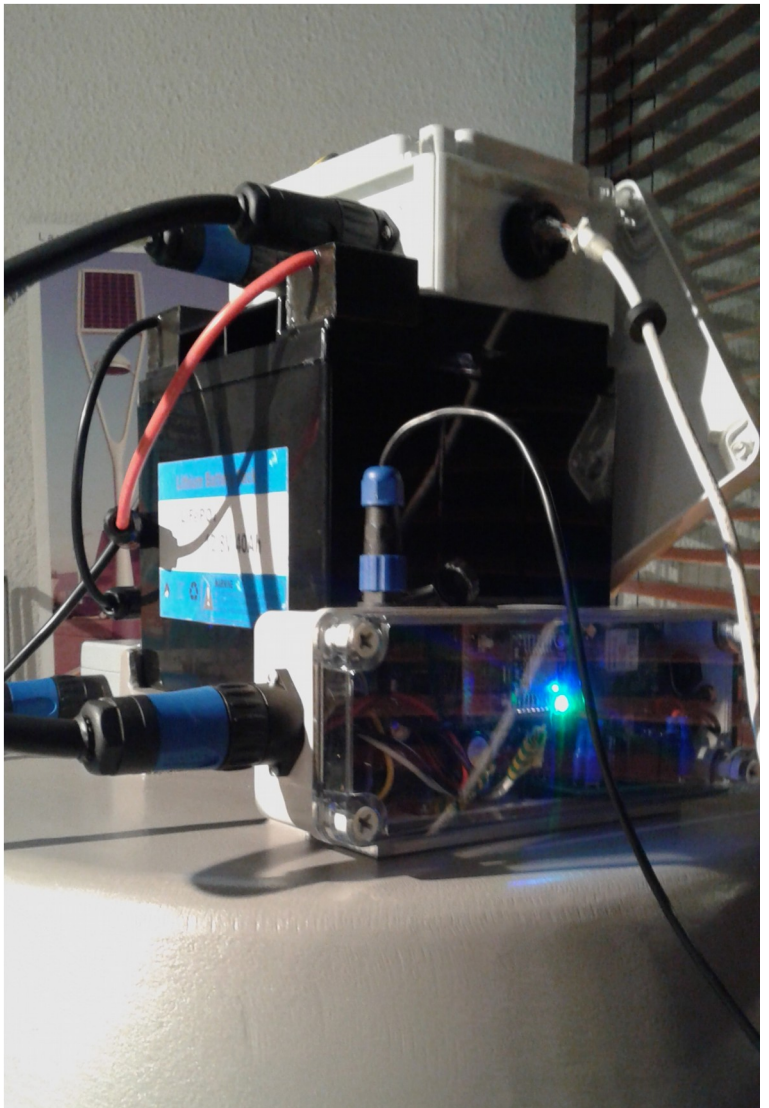
Connexió amb altre maquinari



Comunicacions amb Raspberry Pi

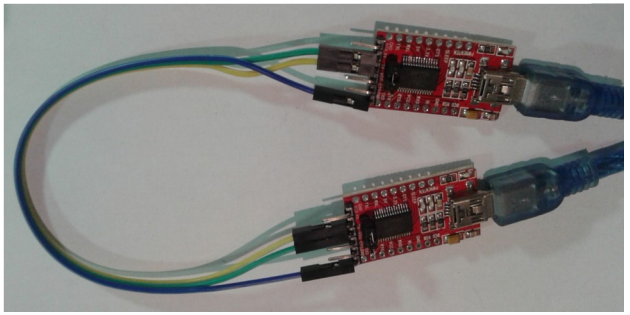
Connexió amb altre maquinari

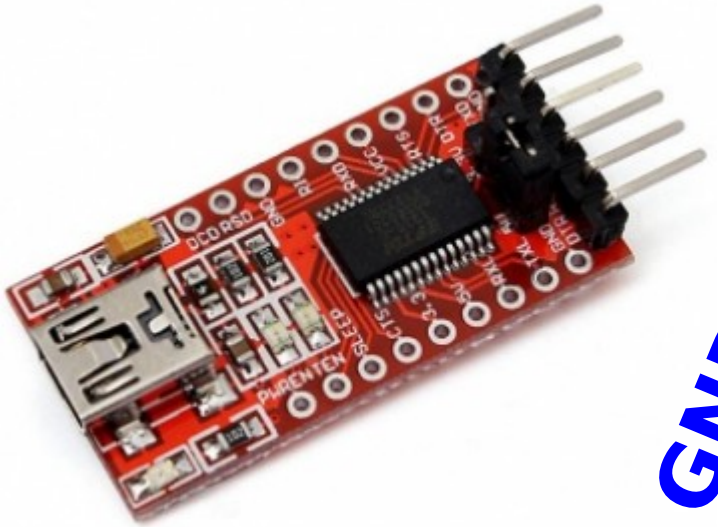


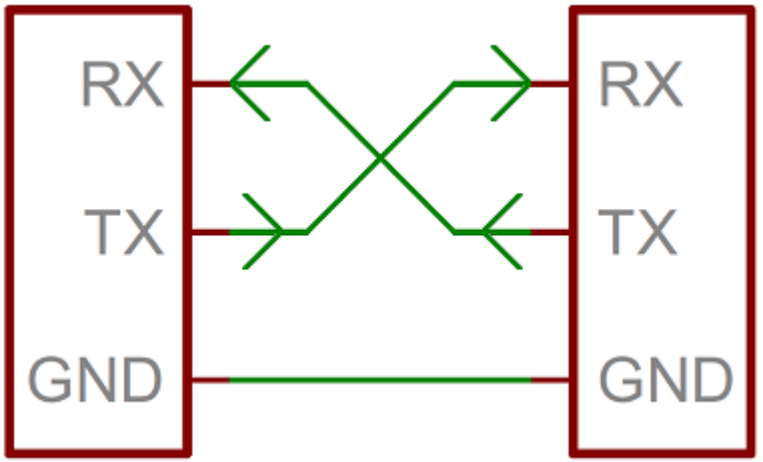


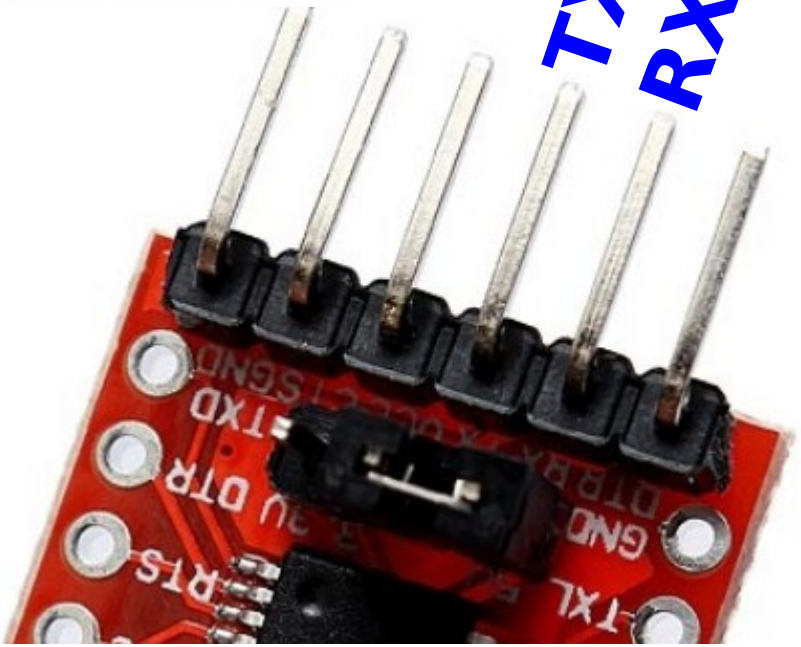
Comunicacions amb Raspberry Pi

Connexió amb altre maquinari



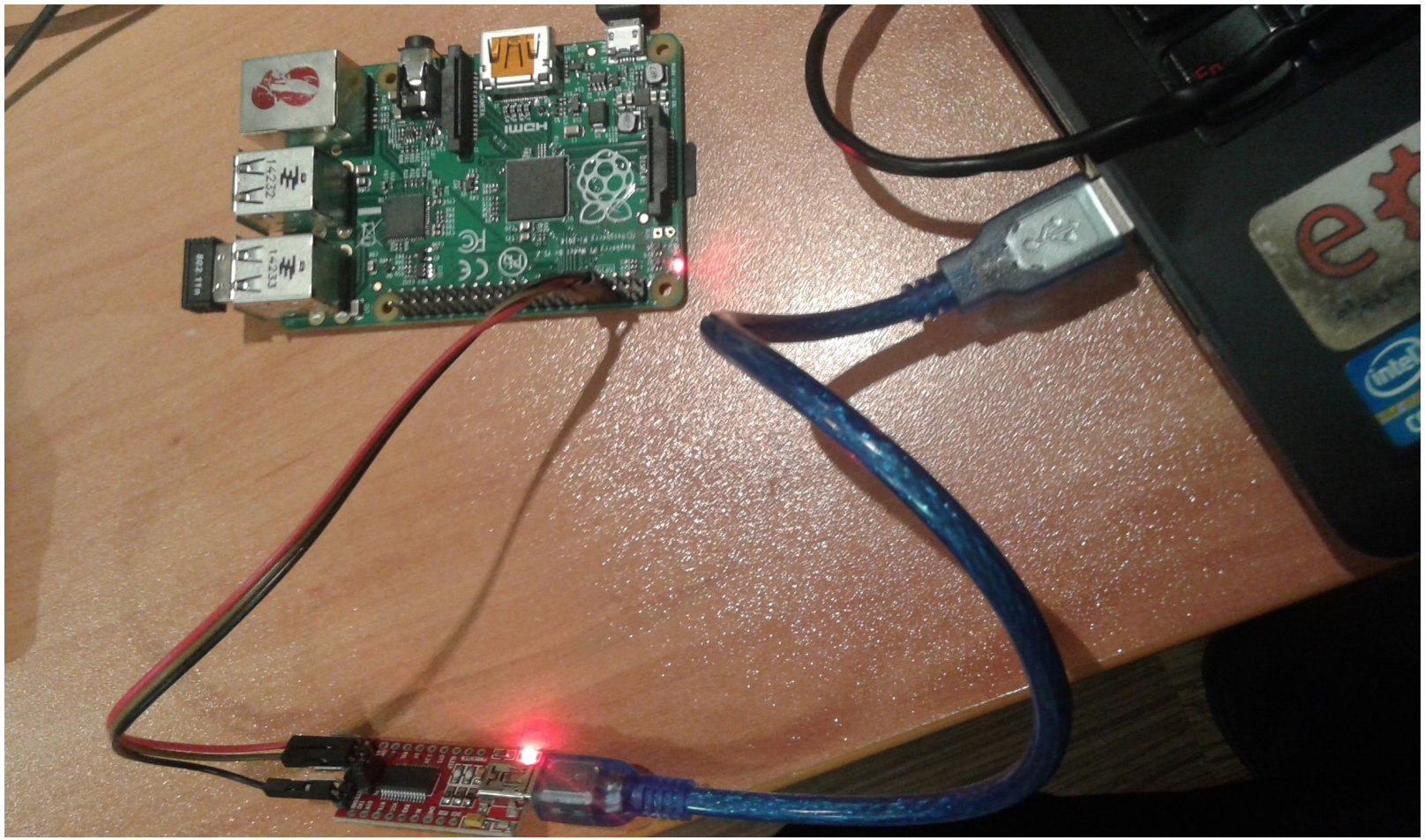






Comunicacions amb Raspberry Pi

Connexió amb altre maquinari



Comunicacions amb Raspberry Pi

Configuració de xarxa

WiFi : https://www.raspberrypi.org/documentation/configuration/wireless/wireless-cli.md

Ethernet : http://hectorgarciaperez.com/2013/05/01/configurar-interfaces-de-red-en-debian-6/

eth0 amb IP estàtica

```
# interfaces(5) file used by ifup(8) and ifdown(8)

# Please note that this file is written to be used with dhcpcd
# For static IP, consult /etc/dhcpcd.conf and 'man dhcpcd.conf'

# Include files from /etc/network/interfaces.d:
source-directory /etc/network/interfaces.d

auto lo
iface lo inet loopback

#eth0 amb DHCP
#allow-hotplug eth0
#iface eth0 inet dhcp

#eth0 amb IP estàtica
allow-hotplug eth0
iface eth0 inet static
    address 192.168.1.2
    netmask 255.255.255.0
    network 192.168.1.0
    broadcast 192.168.1.255
    gateway 192.168.1.1
    dns-nameservers 80.58.0.33 80.58.32.97

#WIFI DHCP
allow-hotplug wlan0
iface wlan0 inet dhcp
    wpa-conf /etc/wpa_supplicant/wpa_supplicant.conf
```

eth0 amb IP dinàmica

```
# interfaces(5) file used by ifup(8) and ifdown(8)

# Please note that this file is written to be used with dhcpcd
# For static IP, consult /etc/dhcpcd.conf and 'man dhcpcd.conf'

# Include files from /etc/network/interfaces.d:
source-directory /etc/network/interfaces.d

auto lo
iface lo inet loopback

#eth0 amb DHCP
allow-hotplug eth0
iface eth0 inet dhcp

#eth0 amb IP estàtica
#allow-hotplug eth0
#iface eth0 inet static
#   address 192.168.1.2
#   netmask 255.255.255.0
#   network 192.168.1.0
#   broadcast 192.168.1.255
#   gateway 192.168.1.1
#   dns-nameservers 80.58.0.33 80.58.32.97

#WIFI DHCP
allow-hotplug wlan0
iface wlan0 inet dhcp
    wpa-conf /etc/wpa_supplicant/wpa_supplicant.conf
```

/etc/network/interfaces

Comunicacions amb Raspberry Pi

Configuració de xarxa

Si modifiqueu **wlan0** a /etc/network/interfaces i no voleu reiniciar l'equip, executeu:

```
ifdown wlan0
ifup wlan0
```

Si modifiqueu **eth0** a /etc/network/interfaces i no voleu reiniciar l'equip, executeu:

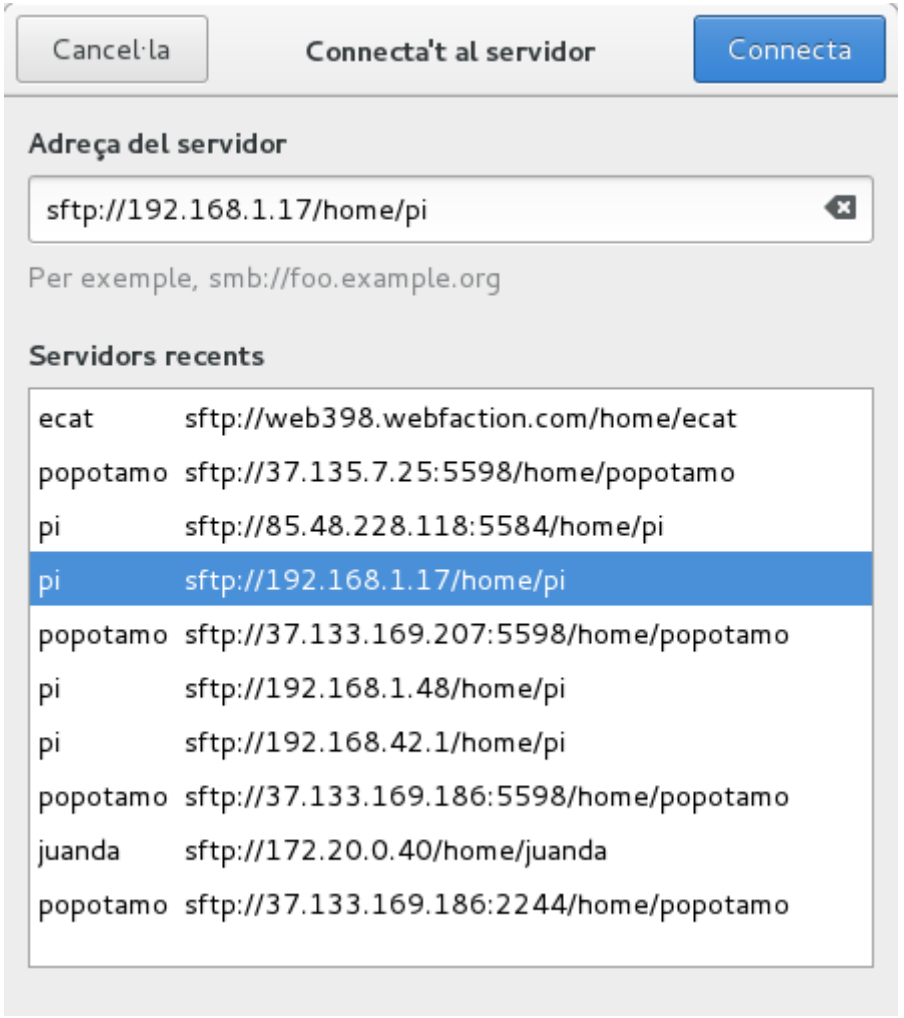
```
ifdown eth0
ifup eth0
```

/etc/wpa_supplicant/wpa_supplicant.conf

```
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
#country=GB

network={
    ssid="MOVISTAR_2840"
    psk="TFTMX3YTUFRC4NAFJUJJ"
}

network={
    ssid="Telecos.cat"
    psk="Fourier17"
    key_mgmt=WPA-PSK
}

network={
    ssid="IoT-eCat"
    psk="clotClot"
    key_mgmt=WPA-PSK
}
```

Comunicacions amb Raspberry Pi

Connexió remota emprant el navegador d'arxius



Comunicacions amb Raspberry Pi

ifconfig

```
pi@raspberrypi:~ $ ifconfig
eth0      Link encap:Ethernet  HWaddr b8:27:eb:fd:d0:b5
          inet addr:192.168.1.17  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::b88e:e7c9:8c24:45d7/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1140 errors:0 dropped:7 overruns:0 frame:0
          TX packets:206 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:77745 (75.9 KiB)  TX bytes:28255 (27.5 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:226 errors:0 dropped:0 overruns:0 frame:0
          TX packets:226 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:20208 (19.7 KiB)  TX bytes:20208 (19.7 KiB)

wlan0     Link encap:Ethernet  HWaddr b8:27:eb:a8:85:e0
          inet addr:192.168.42.1  Bcast:192.168.42.255  Mask:255.255.255.0
          inet6 addr: fe80::4504:2549:cd46:698/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:246 errors:0 dropped:168 overruns:0 frame:0
          TX packets:85 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:39016 (38.1 KiB)  TX bytes:17059 (16.6 KiB)

pi@raspberrypi:~ $
```

Comunicacions amb Raspberry Pi

Porta d'enllaç i DNS

```
pi@raspberrypi:~ $ route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         192.168.1.1     0.0.0.0         UG    202    0        0 eth0
169.254.0.0     0.0.0.0         255.255.0.0     U     303    0        0 wlan0
192.168.1.0     0.0.0.0         255.255.255.0   U     202    0        0 eth0
192.168.42.0    0.0.0.0         255.255.255.0   U     0      0        0 wlan0
pi@raspberrypi:~ $ cat /etc/resolv.conf
# Generated by resolvconf
nameserver 62.81.29.254
nameserver 62.81.16.213
pi@raspberrypi:~ $
```

Comunicacions amb Raspberry Pi

SSH

ssh -X pi@192.168.1.17

```
jordi@debianJB:~$ ssh -X pi@192.168.1.17
pi@192.168.1.17's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Nov 30 10:09:47 2016 from 192.168.1.12
pi@raspberrypi:~ $
```

L'opció -X ens permet poder executar remotament programes amb interfície gràfica d'usuari.

En sistemes operatius Windows existeix el programa putty (http://www.putty.org/)

Comunicacions amb Raspberry Pi

Accés SSH sense contrasenya

https://docs.webfaction.com/user-guide/access.html

Genereu una clau al vostre ordinador (si no existeix `~/.ssh/id_rsa.pub`):

1- Obriu una sessió de terminal.
2- Creeu la carpeta `~/.ssh`, en cas de que no existeixi. (`mkdir -p $HOME/.ssh`)
3- Aneu al directori `~/.ssh` (`cd ~/.ssh` i premeu `Enter`)
4- Genereu les vostres claus (`ssh-keygen -t rsa`)
5- Premeu `Enter` per a contestar totes les preguntes per defecte.

Desplegueu la clau a la Raspberry Pi

1- Copieu la clau a la Raspberry Pi. Entreu `scp ~/.ssh/id_rsa.pub pi@192.168.1.17:temp_id_rsa_key.pub (canvieu 192.168.1.17 per la IP de la vostra Raspberry Pi) i premeu Enter.`
Entreu la contrasenya quan us ho demani (la contrasenya per defecte és `raspberry`).

```
jordi@debianJB:~$ scp ~/.ssh/id_rsa.pub pi@192.168.1.17:temp_id_rsa_key.pub
pi@192.168.1.17's password:
id_rsa.pub                                    100%  396     0.4KB/s   00:00
jordi@debianJB:~$
```

Comunicacions amb Raspberry Pi

Accés SSH sense contrasenya

Desplegueu la clau a la Raspberry Pi

2- Obriu una sessió SSH a la vostra Raspberry Pi des del vostre ordinador. Entreu **`ssh pi@192.168.1.17`** (canvieu 192.168.1.17 per la IP de la vostra Raspberry Pi) i premeu Enter. Entreu la contrasenya quan us ho demani (la contrasenya per defecte és **raspberry**).

```
jordi@debianJB:~$ ssh pi@192.168.1.17
pi@192.168.1.17's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Nov 30 10:18:28 2016 from 192.168.1.12
pi@raspberrypi:~ $
```

3- Creeu la carpeta `~/.ssh`, en cas de que no existeixi. (`mkdir -p $HOME/.ssh`)

4- Afegiu la vostra clau a l'arxiu authorized_keys . Entreu **`cat ~/temp_id_rsa_key.pub >> ~/.ssh/authorized_keys`** i premeu Enter.

```
pi@raspberrypi:~ $ cat ~/temp_id_rsa_key.pub >> ~/.ssh/authorized_keys
```

Comunicacions amb Raspberry Pi

Accés SSH sense contrasenya

Desplegueu la clau a la Raspberry Pi

5- Esborreu l'arxiu amb la clau tempral. Feu `rm temp_id_rsa_key.pub` i premeu `Enter`.

```
pi@raspberrypi:~ $ rm temp_id_rsa_key.pub
```

6- Protegiu l'arxiu de claus SSH. Entreu-hi `chmod 600 ~/.ssh/authorized_keys` i premeu `Enter`.

```
pi@raspberrypi:~ $ chmod 600 ~/.ssh/authorized keys
```

7- Protegiu el directori SSH. Entreu-hi `chmod 700 ~/.ssh` i premeu `Enter`.

```
pi@raspberrypi:~ $ chmod 700 ~/.ssh
```

8- Protegiu el vostre directori d'usuari. Entreu-hi `chmod go-w $HOME` i premeu `Enter`.

```
pi@raspberrypi:~ $ chmod go-w $HOME
```

9- Tanqueu la sessió SSH.

Comunicacions amb Raspberry Pi

Accés SSH sense contrasenya

Desplegueu la clau a la Raspberry Pi

```
pi@raspberrypi:~ $ cat ~/temp_id_rsa_key.pub >> ~/.ssh/authorized_keys
pi@raspberrypi:~ $ rm temp_id_rsa_key.pub
pi@raspberrypi:~ $ chmod 600 ~/.ssh/authorized_keys
pi@raspberrypi:~ $ chmod 700 ~/.ssh
pi@raspberrypi:~ $ chmod go-w $HOME
pi@raspberrypi:~ $ exit
logout
Connection to 192.168.1.17 closed.
jordi@debianJB:~$
```

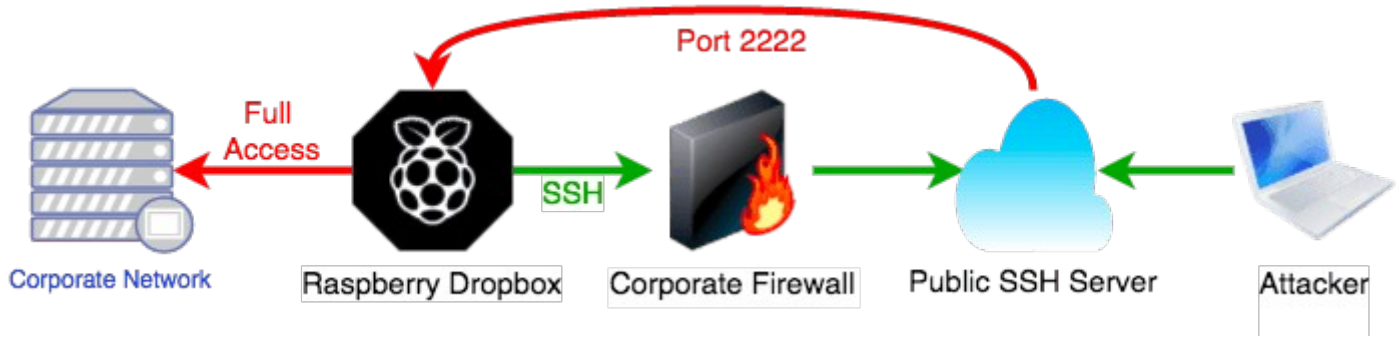
Verifiqueu que ja podeu accedir mitjançant SSH sense que us pregunti la contrasenya

```
jordi@debianJB:~$ ssh pi@192.168.1.17

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Nov 30 11:03:54 2016 from 192.168.1.12
pi@raspberrypi:~ $
```

Comunicacions amb Raspberry Pi

Túnel SSH invers

http://jerrygamblin.com/2016/04/23/persistent-reverse-ssh-tunnels-on-a-raspberrypi/



Comunicacions amb Raspberry Pi

Túnel SSH invers

jordi@debianJB: ~

Fitxer Edita Visualitza Cerca Terminal Ajuda

```
pi@raspberrypi:~ $ ssh -R 12345:localhost:22 jordi@192.168.1.12
The authenticity of host '192.168.1.12 (192.168.1.12)' can't be established.
ECDSA key fingerprint is 01:eb:89:96:25:8b:48:22:e9:a7:ed:5e:d4:98:c4:ac.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.12' (ECDSA) to the list of known hosts.
jordi@192.168.1.12's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes wi
permitted by applicable l
jordi@debianJB:~$
```

pi@raspberrypi: ~

Fitxer Edita Visualitza Cerca Terminal Ajuda

```
jordi@debianJB:~$ ssh -p 12345 pi@127.0.0.1
The authenticity of host '[127.0.0.1]:12345 ([127.0.0.1]:12345)' can't be establ
ished.
ECDSA key fingerprint is db:79:95:3e:5d:97:96:cd:ce:57:64:96:63:82:15:e2.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[127.0.0.1]:12345' (ECDSA) to the list of known host
s.

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Nov 30 23:46:52 2016 from 192.168.1.12
pi@raspberrypi:~ $
```

Comunicacions amb Raspberry Pi

autoSSH

Si el túnel deixa de funcionar (acostuma a ocórrer en xarxes molt ocupades o d'escasa qualitat, com el 3G), es pot utilitzar el paquet autossh en lloc de l'ssh per a establir la connexió que s'encarregarà de mantenir el túnel obert reiniciant automàticament la connexió.

```
pi@raspberrypi:~ $ autossh -M 65500 -o ServerAliveInterval=20 -R 19994:localhost:22 ecat@web398.webfaction.com
```

L'autossh no funciona com s'espera si al connectar la Raspberry Pi no hi ha connexió a Internet. Per a evitar això, primer verifiquem la connectivitat fent un ping.

```
jordi@debianJB:~$ ssh ecat@web398.webfaction.com
ecat@web398.webfaction.com's password:
Last login: Wed Nov 30 23:21:27 2016 from 62.83.215.143
[ecat@web398 ~]$ ssh -p 19994 pi@localhost
pi@localhost's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Dec  1 00:24:02 2016 from localhost
pi@raspberrypi:~ $
```

Comunicacions amb Raspberry Pi

Verificació de la connectivitat

```
pi@raspberrypi:~ $ ping -c 1 www.binefa.cat
PING binefa.cat (37.58.75.228) 56(84) bytes of data.
64 bytes from web398.webfaction.com (37.58.75.228): icmp_seq=1 ttl=52 time=49.2 ms

--- binefa.cat ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 49.200/49.200/49.200/0.000 ms
pi@raspberrypi:~ $ echo $?
0
pi@raspberrypi:~ $ ping -c 1 www.jordibinefa.cat
ping: unknown host www.jordibinefa.cat
pi@raspberrypi:~ $ echo $?
2
pi@raspberrypi:~ $ ping -c 1 192.168.1.222
PING 192.168.1.222 (192.168.1.222) 56(84) bytes of data.
From 192.168.1.17 icmp_seq=1 Destination Host Unreachable

--- 192.168.1.222 ping statistics ---
1 packets transmitted, 0 received, +1 errors, 100% packet loss, time 0ms

pi@raspberrypi:~ $ echo $?
1
pi@raspberrypi:~ $ ping -c 1 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=1.01 ms

--- 192.168.1.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 1.019/1.019/1.019/0.000 ms
pi@raspberrypi:~ $ echo $?
0
pi@raspberrypi:~ $
```

Comunicacions amb Raspberry Pi

Ordres consecutives emprant &&

```
pi@raspberrypi:~ $ ping -c 1 192.168.1.222 && ls
PING 192.168.1.222 (192.168.1.222) 56(84) bytes of data.
From 192.168.1.17 icmp_seq=1 Destination Host Unreachable

--- 192.168.1.222 ping statistics ---
1 packets transmitted, 0 received, +1 errors, 100% packet loss, time 0ms

pi@raspberrypi:~ $ ping -c 1 192.168.1.1 && ls
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=1.01 ms

--- 192.168.1.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 1.019/1.019/1.019/0.000 ms
codis    Documents  Music         Pictures  pwdVnc.txt    sketchbook  tunelSSH.txt
Desktop  Downloads  oldconffiles  Public    python_games  Templates   Videos
pi@raspberrypi:~ $ ping -c 1 www.jordibinefa.cat && ls
ping: unknown host www.jordibinefa.cat
pi@raspberrypi:~ $ ping -c 1 www.binefa.cat && ls
PING binefa.cat (37.58.75.228) 56(84) bytes of data.
64 bytes from web398.webfaction.com (37.58.75.228): icmp_seq=1 ttl=52 time=48.6 ms

--- binefa.cat ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 48.642/48.642/48.642/0.000 ms
codis    Documents  Music         Pictures  pwdVnc.txt    sketchbook  tunelSSH.txt
Desktop  Downloads  oldconffiles  Public    python_games  Templates   Videos
pi@raspberrypi:~ $
```

En cas de que la primera ordre tingui èxit, llavors executa la següent.

Comunicacions amb Raspberry Pi

Processos actius i grep

```
pi@raspberrypi:~ $ ps aux | grep autossh
pi        2586  0.0  0.2   4772  2032 pts/0    S+   12:18   0:00 grep --color=auto autossh
pi@raspberrypi:~ $ ps aux | grep autossh | grep -v grep
pi@raspberrypi:~ $ if ! ps aux | grep autossh > /dev/null; then echo "No hi ha cap procés"; else  echo "Hi ha algun procés"; fi
Hi ha algun procés
pi@raspberrypi:~ $ if ! ps aux | grep autossh | grep -v grep > /dev/null; then echo "No hi ha cap procés"; else  echo "Hi ha algun pro
cés"; fi
No hi ha cap procés
pi@raspberrypi:~ $ if ! ps aux | grep autossh > /dev/null; then echo "No hi ha cap procés"; fi
pi@raspberrypi:~ $ if ! ps aux | grep autossh | grep -v grep > /dev/null; then echo "No hi ha cap procés"; fi
No hi ha cap procés
pi@raspberrypi:~ $
```

Comunicacions amb Raspberry Pi

crontab

/etc/crontab

```
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
*/1 *   * * *   root      ping -c 1 www.binefa.cat && if ! ps aux | grep autossh | grep -v grep > /dev/nu
#
```

```
*/1 *   * * *   root      ping -c 1 www.binefa.cat
&& if ! ps aux | grep autossh | grep -v grep >
/dev/null; then su pi -c 'autossh -f -nNT -M 65500
-o ServerAliveInterval=20 -R 19994:localhost:22
ecat@web398.webfaction.com'; fi
```

Comunicacions amb Raspberry Pi

Persistència de les accions remotes

screen (1/2)

http://www.tecmint.com/screen-command-examples-to-manage-linux-terminals/

```
pi@raspberrypi:~ $ screen -ls
No Sockets found in /var/run/screen/S-pi.

pi@raspberrypi:~ $ screen
```

```
pi@raspberrypi:~ $ ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=2.07 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=0.919 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=1.17 ms
64 bytes from 192.168.1.1: icmp_seq=4 ttl=64 time=0.960 ms
64 bytes from 192.168.1.1: icmp_seq=5 ttl=64 time=0.934 ms
64 bytes from 192.168.1.1: icmp_seq=6 ttl=64 time=1.33 ms
```

Per a sortir temporalment d'screen premeu **Ctrl A + D**

```
[detached from 2427.pts-0.raspberrypi]
pi@raspberrypi:~ $
```

Comunicacions amb Raspberry Pi

Persistència de les accions remotes screen (2/2)

http://www.tecmint.com/screen-command-examples-to-manage-linux-terminals/

```
[detached from 2427.pts-0.raspberrypi]
pi@raspberrypi:~ $ exit
logout
Connection to 192.168.1.17 closed.
jordi@debianJB:~$ ssh pi@192.168.1.17

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Nov 30 11:55:07 2016 from 192.168.1.12
pi@raspberrypi:~ $ screen -r

64 bytes from 192.168.1.1: icmp_seq=205 ttl=64 time=0.883 ms
64 bytes from 192.168.1.1: icmp_seq=206 ttl=64 time=0.881 ms
64 bytes from 192.168.1.1: icmp_seq=207 ttl=64 time=0.969 ms
64 bytes from 192.168.1.1: icmp_seq=208 ttl=64 time=0.873 ms
64 bytes from 192.168.1.1: icmp_seq=209 ttl=64 time=1.10 ms
64 bytes from 192.168.1.1: icmp_seq=210 ttl=64 time=0.927 ms
64 bytes from 192.168.1.1: icmp_seq=211 ttl=64 time=0.906 ms
64 bytes from 192.168.1.1: icmp_seq=212 ttl=64 time=0.860 ms


[screen is terminating]
pi@raspberrypi:~ $
```

Per a cancel·lar el ping podem prémer Ctrl C i per a sortir de `screen` fem `exit`

Comunicacions amb Raspberry Pi

Publicació de la IP pública (1/2)


```
popotamo@popotamo:~$ cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
#
*/5 *   * * *   popotamo /home/popotamo/bin/svr.sh
#
*/1 *   * * *   popotamo /home/popotamo/bin/checkAutoSSH.sh
popotamo@popotamo:~$
```

Comunicacions amb Raspberry Pi

Publicació de la IP pública (2/2)

```
popotamo@popotamo:~$ cat /home/popotamo/bin/svr.sh
#!/bin/bash

# rm ip.php
wget http://www.binefa.cat/php/ip/ip.php
echo "<BR>" >> ip.php
date >> ip.php

scp ip.php ecat@web398.webfaction.com:/home/ecat/webapps/www_binefa_cat_php/svr/index.php

rm ip.php
popotamo@popotamo:~$
```

ui_ctrlLeds01.py | ecat.cpp | index.htm | index.php

```
1  <META HTTP-EQUIV="refresh" CONTENT="0;URL=ip.php">
2
```

ecat.cpp | index.htm | index.php | ip.php

```
1  <?php
2      echo $_SERVER['REMOTE_ADDR'];
3  ?>
4
```

ecat.cpp | index.htm | index.php | ip.php

```
1  <HTML>
2      <HEAD>
3          <TITLE>Your Internet IP</TITLE>
4      </HEAD>
5      <BODY>
6          <?php
7              echo $_SERVER['REMOTE_ADDR'];
8          ?>
9      </BODY>
10 </HTML>
11
```

Comunicacions amb Raspberry Pi

Raspberry Pi com a punt d'accés (1/4)

http://elinux.org/RPI-Wireless-Hotspot

Instal·lació de paquets:

sudo apt-get install hostapd udhcpd

```
pi@raspberrypi:~ $ cat /etc/udhcpd.conf
# Sample udhcpd configuration file (/etc/udhcpd.conf)

# The start and end of the IP lease block

start           192.168.42.20   #default: 192.168.0.20
end             192.168.42.254  #default: 192.168.0.254


# The interface that udhcpd will use

interface       wlan0           #default: eth0


#Examles
#opt    dns     192.168.10.2 192.168.10.10
opt     dns     8.8.8.8 4.2.2.2
option  subnet  255.255.255.0
#opt    router  192.168.10.2
opt     router  192.168.42.1
#opt    wins    192.168.10.10
#option dns     129.219.13.81   # appened to above DNS servers for a total of 3
#option domain  local
option  lease   864000          # 10 days of seconds
```

Comunicacions amb Raspberry Pi

Raspberry Pi com a punt d'accés (2/4)

```
pi@raspberrypi:~ $ cat /etc/default/udhcpd
# Comment the following line to enable
#DHCPD_ENABLED="no"

# Options to pass to busybox' udhcpd.
#
# -S    Log to syslog
# -f    run in foreground

DHCPD_OPTS="-S"
pi@raspberrypi:~ $
```

```
pi@raspberrypi:~ $ cat /etc/network/interfaces
# interfaces(5) file used by ifup(8) and ifdown(8)

# Please note that this file is written to be used with dhcpcd
# For static IP, consult /etc/dhcpcd.conf and 'man dhcpcd.conf'

# Include files from /etc/network/interfaces.d:
source-directory /etc/network/interfaces.d

auto lo
iface lo inet loopback

iface eth0 inet manual

allow-hotplug wlan0
#iface wlan0 inet manual
#    wpa-conf /etc/wpa_supplicant/wpa_supplicant.conf

iface wlan0 inet static
    address 192.168.42.1
    netmask 255.255.255.0


allow-hotplug wlan1
iface wlan1 inet manual
    wpa-conf /etc/wpa_supplicant/wpa_supplicant.conf

up iptables-restore < /etc/iptables.ipv4.nat
```

Comunicacions amb Raspberry Pi

Raspberry Pi com a punt d'accés (3/4)

```
pi@raspberrypi:~ $ cat /etc/hostapd/hostapd.conf
interface=wlan0
driver=nl80211
ssid=pvSL
hw_mode=g
channel=10
macaddr_acl=0
auth_algs=1
ignore_broadcast_ssid=0
wpa=2
wpa_passphrase=fanalBolet0
wpa_key_mgmt=WPA-PSK
#wpa_pairwise=TKIP      # You better do not use this weak encryption (only used by old client devices)
rsn_pairwise=CCMP
ieee80211n=1           # 802.11n support
wmm_enabled=1          # QoS support
ht_capab=[HT40][SHORT-GI-20][DSSS_CCK-40]
pi@raspberrypi:~ $
```

```
pi@raspberrypi:~ $ cat /etc/default/hostapd
# Defaults for hostapd initscript
#
# See /usr/share/doc/hostapd/README.Debian for information about
# methods of managing hostapd.
#
# Uncomment and set DAEMON_CONF to the absolute path of a hostap
# file and hostapd will be started during system boot. An exampl
# file can be found at /usr/share/doc/hostapd/examples/hostapd.c
#
DAEMON_CONF="/etc/hostapd/hostapd.conf"

# Additional daemon options to be appended to hostapd command:-
#       -d   show more debug messages (-dd for even more)
#       -K   include key data in debug messages
#       -t   include timestamps in some debug messages
#
# Note that -B (daemon mode) and -P (pidfile) options are automa
# configured by the init.d script and must not be added to DAEMC
#
#DAEMON_OPTS=""
pi@raspberrypi:~ $
```

Comunicacions amb Raspberry Pi

Raspberry Pi com a punt d'accés (4/4)

A l'arxiu /etc/sysctl.conf

```
# Uncomment the next line to enable packet forwarding for IPv4
net.ipv4.ip_forward=1
```

```
pi@raspberrypi:~ $ cat /etc/iptables.ipv4.nat
# Generated by iptables-save v1.4.21 on Sat Oct  8 22:18:23 2016
*filter
:INPUT ACCEPT [36:2032]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [41:6275]
-A FORWARD -i eth0 -o wlan0 -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -i wlan0 -o eth0 -j ACCEPT
COMMIT
# Completed on Sat Oct  8 22:18:23 2016
# Generated by iptables-save v1.4.21 on Sat Oct  8 22:18:23 2016
*nat
:PREROUTING ACCEPT [0:0]
:INPUT ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
:POSTROUTING ACCEPT [0:0]
-A POSTROUTING -o eth0 -j MASQUERADE
COMMIT
# Completed on Sat Oct  8 22:18:23 2016
pi@raspberrypi:~ $
```

Per a reiniciar els serveis:

sudo update-rc.d hostapd enable
sudo update-rc.d udhcpd enable

Comunicacions amb Raspberry Pi

Configuració de la Raspberry Pi

```
pi@raspberrypi ~ $ sudo raspi-config
```

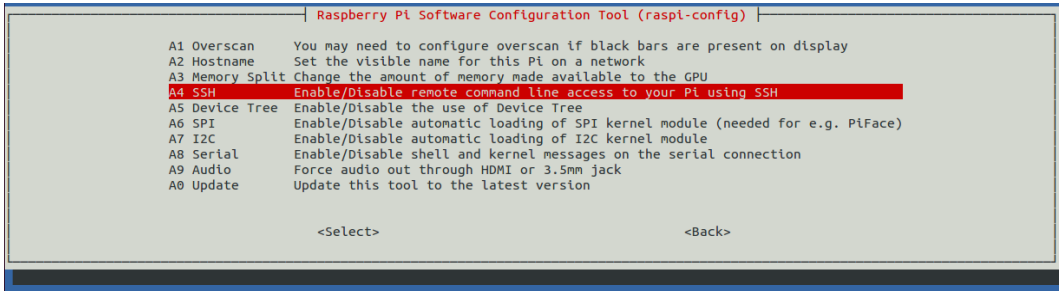
```
                  Raspberry Pi Software Configuration Tool (raspi-config)

    1 Expand Filesystem               Ensures that all of the SD card storage is available to the OS
    2 Change User Password            Change password for the default user (pi)
    3 Enable Boot to Desktop/Scratch  Choose whether to boot into a desktop environment, Scratch, or the command-line
    4 Internationalisation Options    Set up language and regional settings to match your location
    5 Enable Camera                   Enable this Pi to work with the Raspberry Pi Camera
    6 Add to Rastrack                 Add this Pi to the online Raspberry Pi Map (Rastrack)
    7 Overclock                       Configure overclocking for your Pi
    8 Advanced Options                Configure advanced settings
    9 About raspi-config              Information about this configuration tool


                            <Select>                                  <Finish>
```

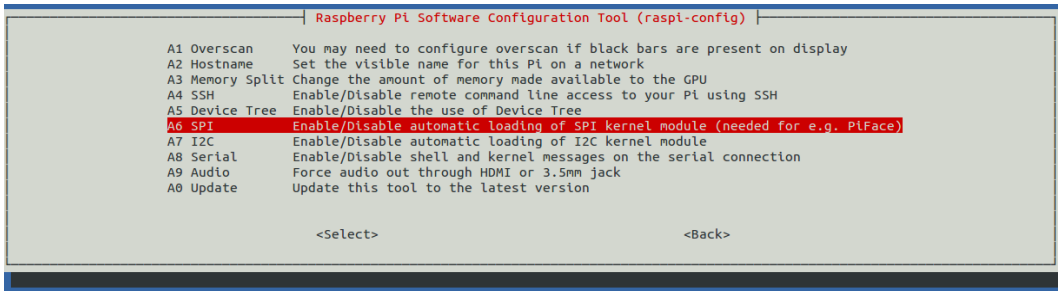
```
 Chose boot option

Console Text console, requiring login (default)
Desktop Log in as user 'pi' at the graphical desktop
Scratch Start the Scratch programming environment upon boot

                <Ok>                <Cancel>
```

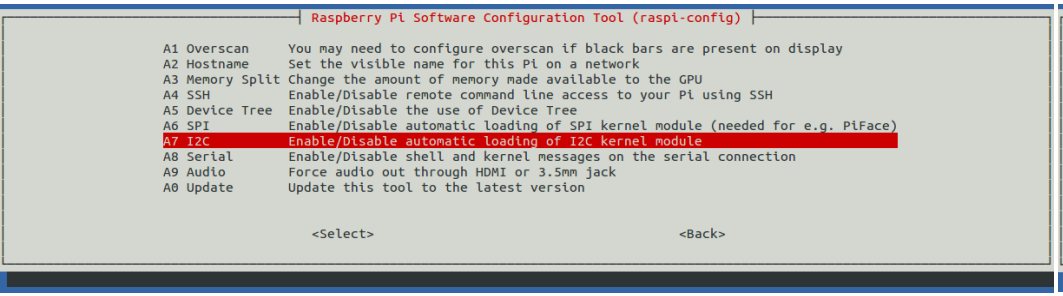
Comunicacions amb Raspberry Pi

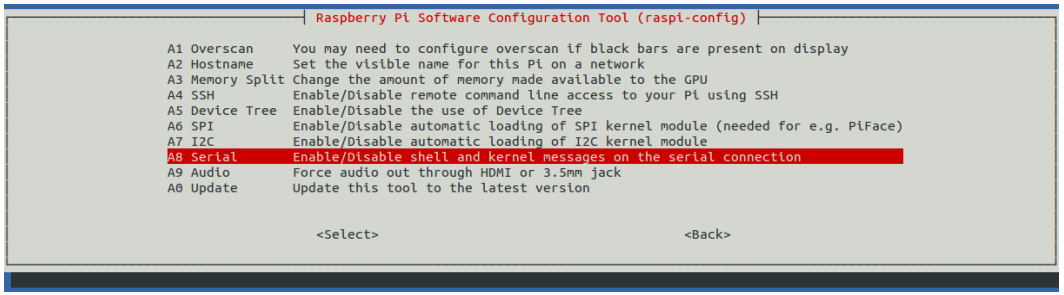
Configuració de la Raspberry Pi

```
pi@raspberrypi ~ $ sudo raspi-config
```

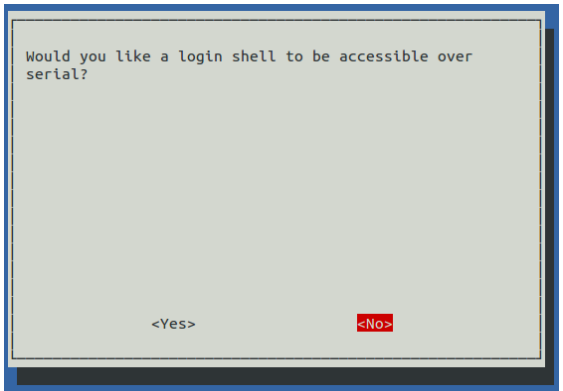






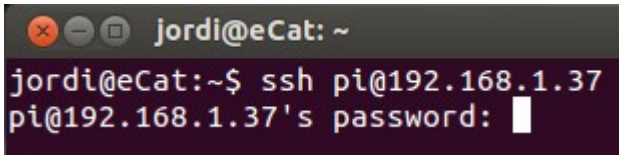


Configuració del port sèrie per a versions més antigues de Raspbian :
http://electronics.cat/doc/3Bpi/rs232/setRs232up.html



Comunicacions amb Raspberry Pi

Accés remot mitjançant SSH




```
jordi@eCat:~$ ssh pi@192.168.1.37
pi@192.168.1.37's password:
Linux raspberrypi 3.18.7+ #755 PREEMPT Thu Feb 12 17:14:31 GMT 2015 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Apr  8 13:03:17 2015 from 192.168.1.5
pi@raspberrypi ~ $
```

Per accedir amb opció d'executar remotament programes amb interfície gràfica (GUI) :

ssh -X pi@192.168.1.37

Comunicacions amb Raspberry Pi

Configuració de l'I2C

```
pi@raspberrypi ~ $ sudo nano /etc/modules
```

```
pi@raspberrypi ~ $ sudo nano /etc/modules
pi@raspberrypi ~ $ cat /etc/modprobe.d/raspi-blacklist.conf
pi@raspberrypi ~ $
```

```
pi@raspberrypi ~ $ cat /etc/modules
# /etc/modules: kernel modules to load at boot time.
#
# This file contains the names of kernel modules that should be
# at boot time, one per line. Lines beginning with "#" are ign
# Parameters can be specified after the module name.

snd-bcm2835
i2c-bcm2708
i2c-dev
```

```
pi@raspberrypi ~ $ uname -a
Linux raspberrypi 3.18.7+ #755 PREEMPT Thu Feb 12 17:14:31 GMT
pi@raspberrypi ~ $ tail /boot/config.txt
#config_hdmi_boost=4

# uncomment for composite PAL
#sdtv_mode=2

#uncomment to overclock the arm. 700 MHz is the default.
#arm_freq=800

dtparam=spi=on
dtparam=i2c_arm=on
pi@raspberrypi ~ $
```

```
pi@raspberrypi ~ $ tail /boot/config.txt
#config_hdmi_boost=4

# uncomment for composite PAL
#sdtv_mode=2

#uncomment to overclock the arm. 700 MHz is the defaul
#arm_freq=800

dtparam=spi=on
dtparam=i2c_arm=on
pi@raspberrypi ~ $ sudo nano /boot/config.txt
pi@raspberrypi ~ $ tail /boot/config.txt

# uncomment for composite PAL
#sdtv_mode=2

#uncomment to overclock the arm. 700 MHz is the defaul
#arm_freq=800

dtparam=spi=on
dtparam=i2c1=on
dtparam=i2c_arm=on
pi@raspberrypi ~ $
```

Comunicacions amb Raspberry Pi

Configuració de l'I2C

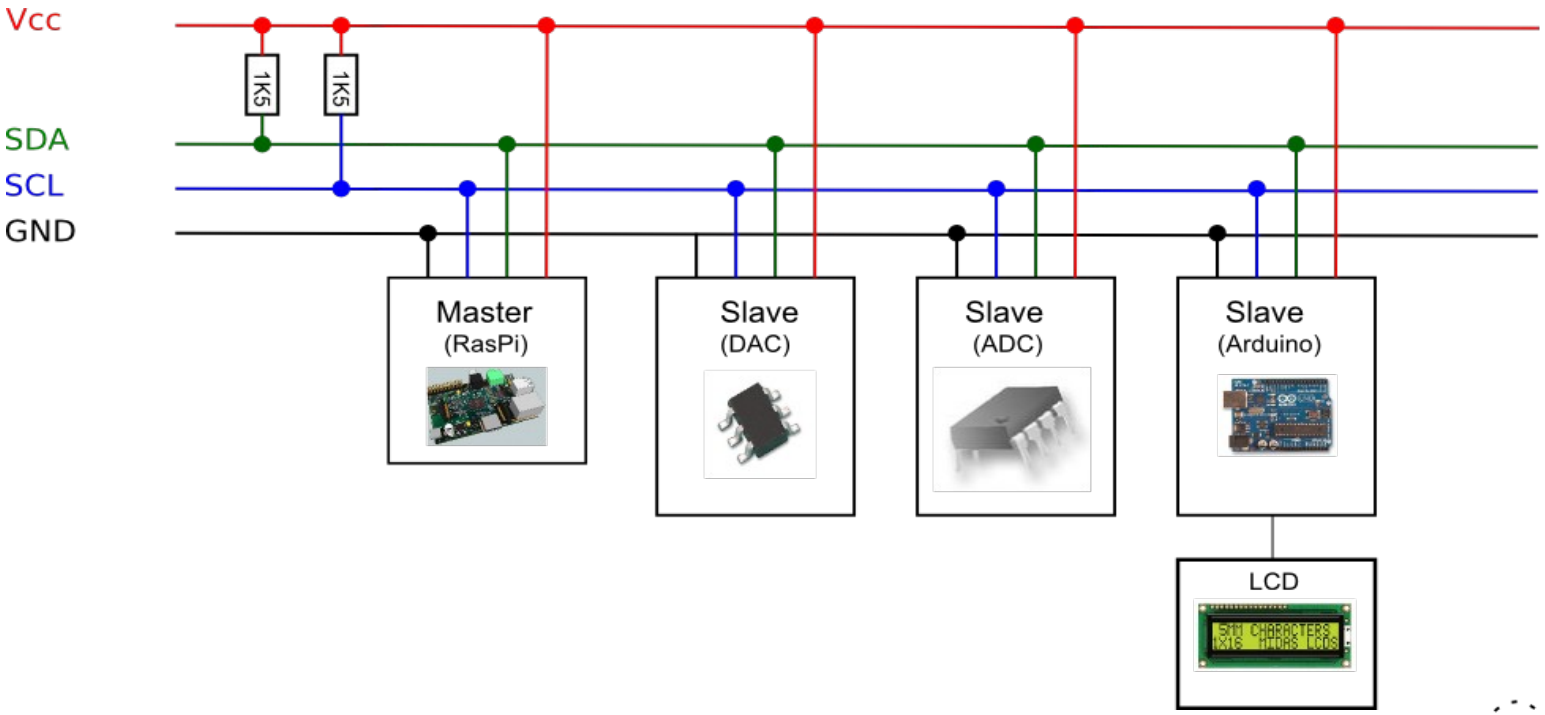
```
pi@raspberrypi ~ $ sudo aptitude install i2c-tools python-smbus
```

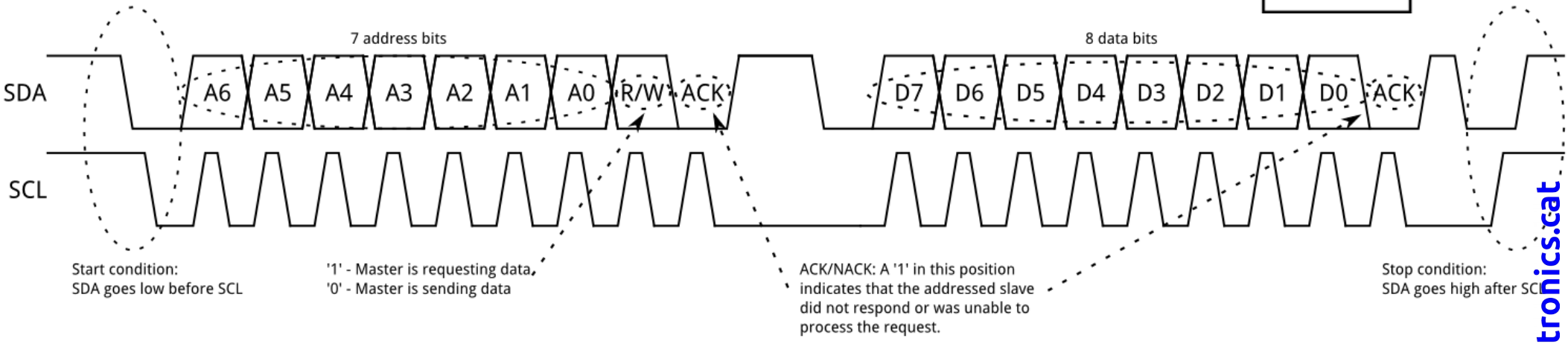
```
pi@raspberrypi:~ $ sudo i2cdetect -y 1
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:          -- -- -- -- -- -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- 38 39 -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
pi@raspberrypi:~ $
```

```
sudo i2cset -y 1 0x38 0xA5
sudo i2cset -y 1 0x39 0xF0
sudo i2cset -y 1 0x39 0xF
```

Comunicacions amb Raspberry Pi

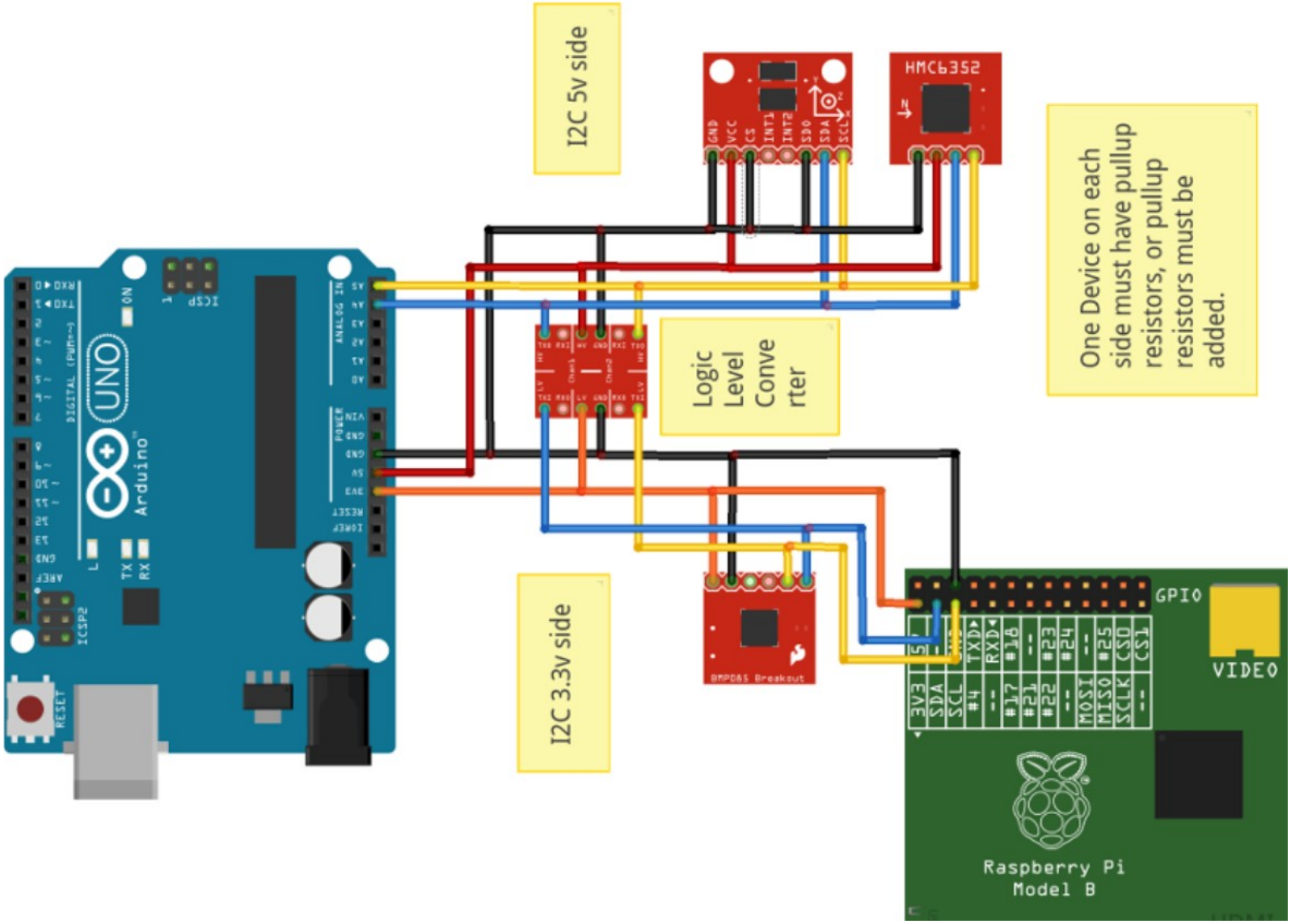
I2C





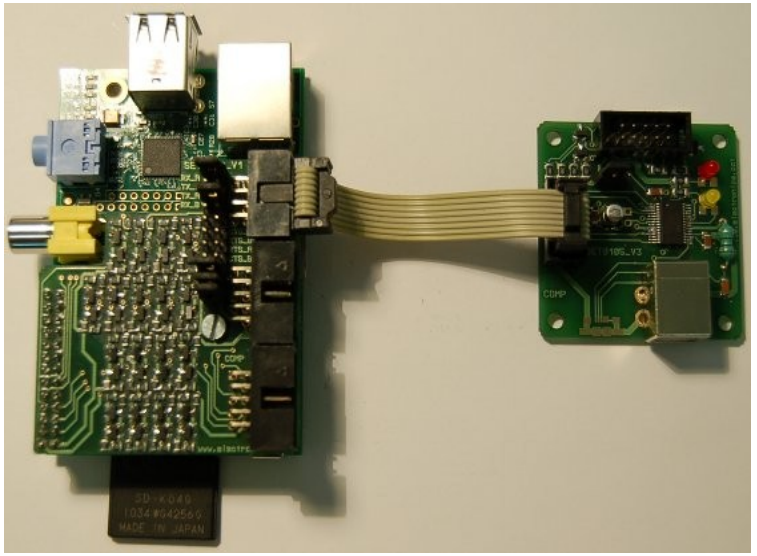
Comunicacions amb Raspberry Pi

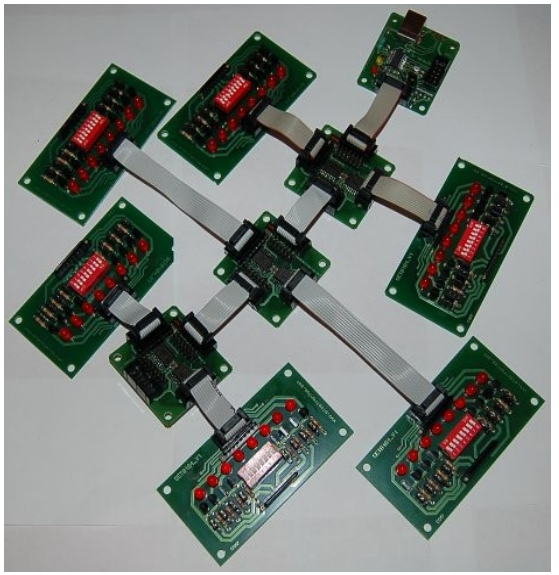
I2C

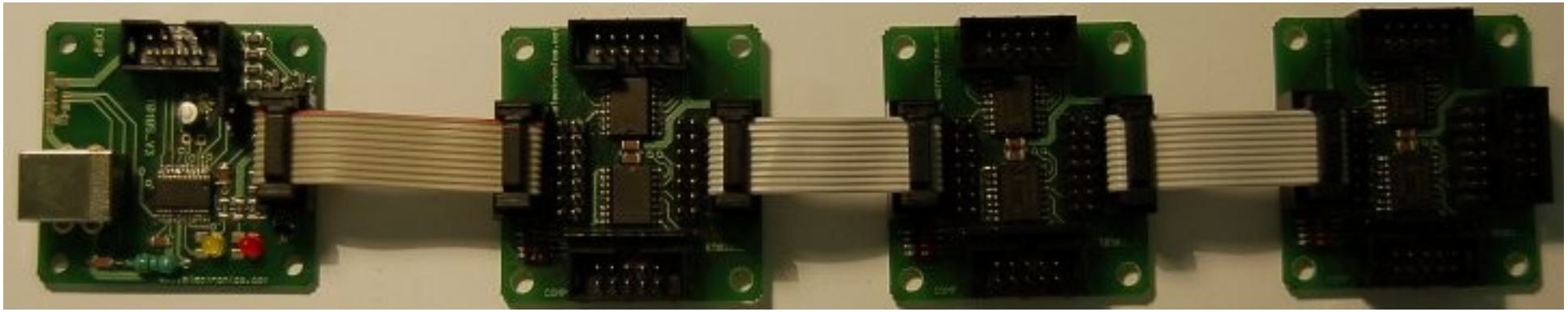


Comunicacions amb Raspberry Pi

I2C



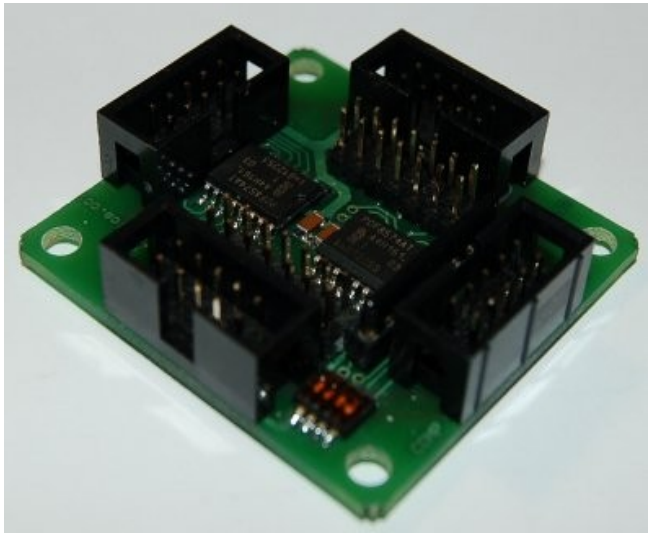


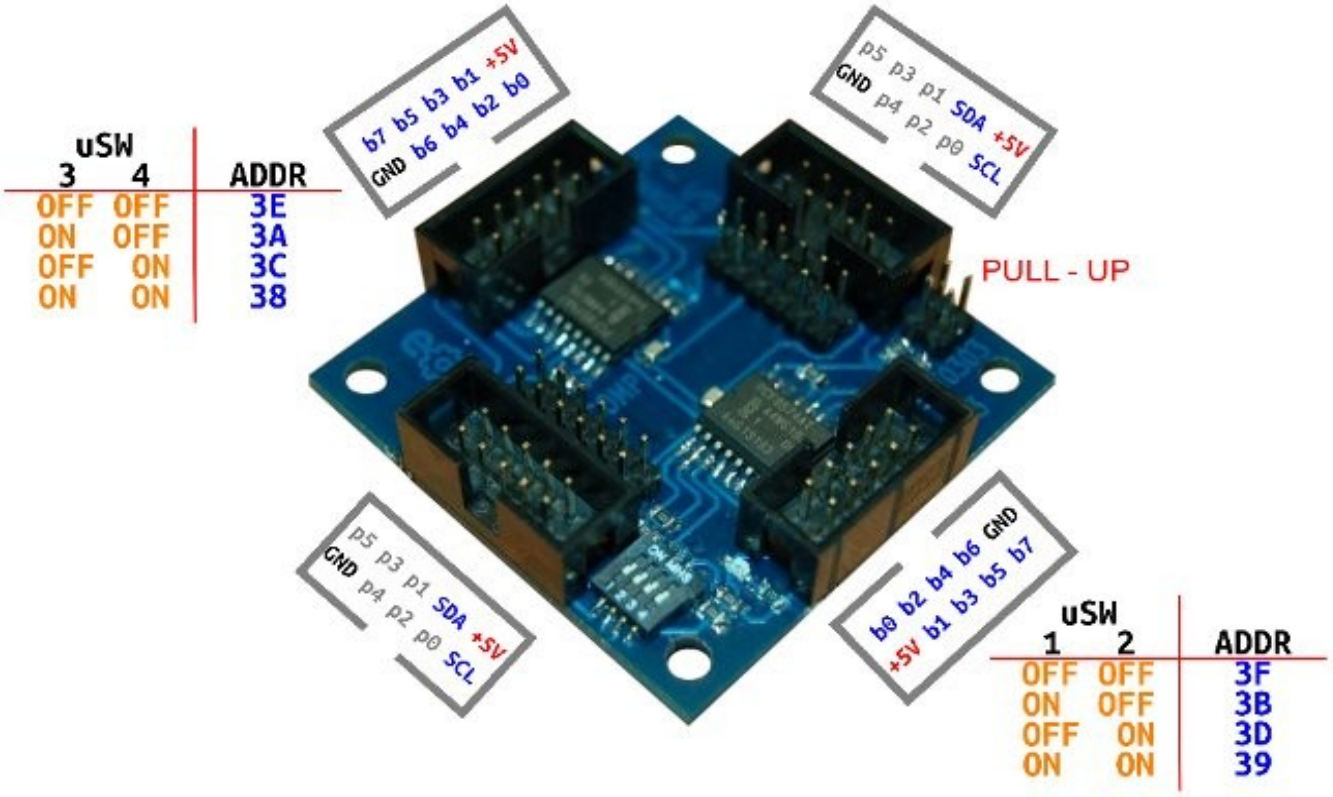


http://electronics.cat/php/common/index.php?lang=ca&page=105

Comunicacions amb Raspberry Pi

I2C





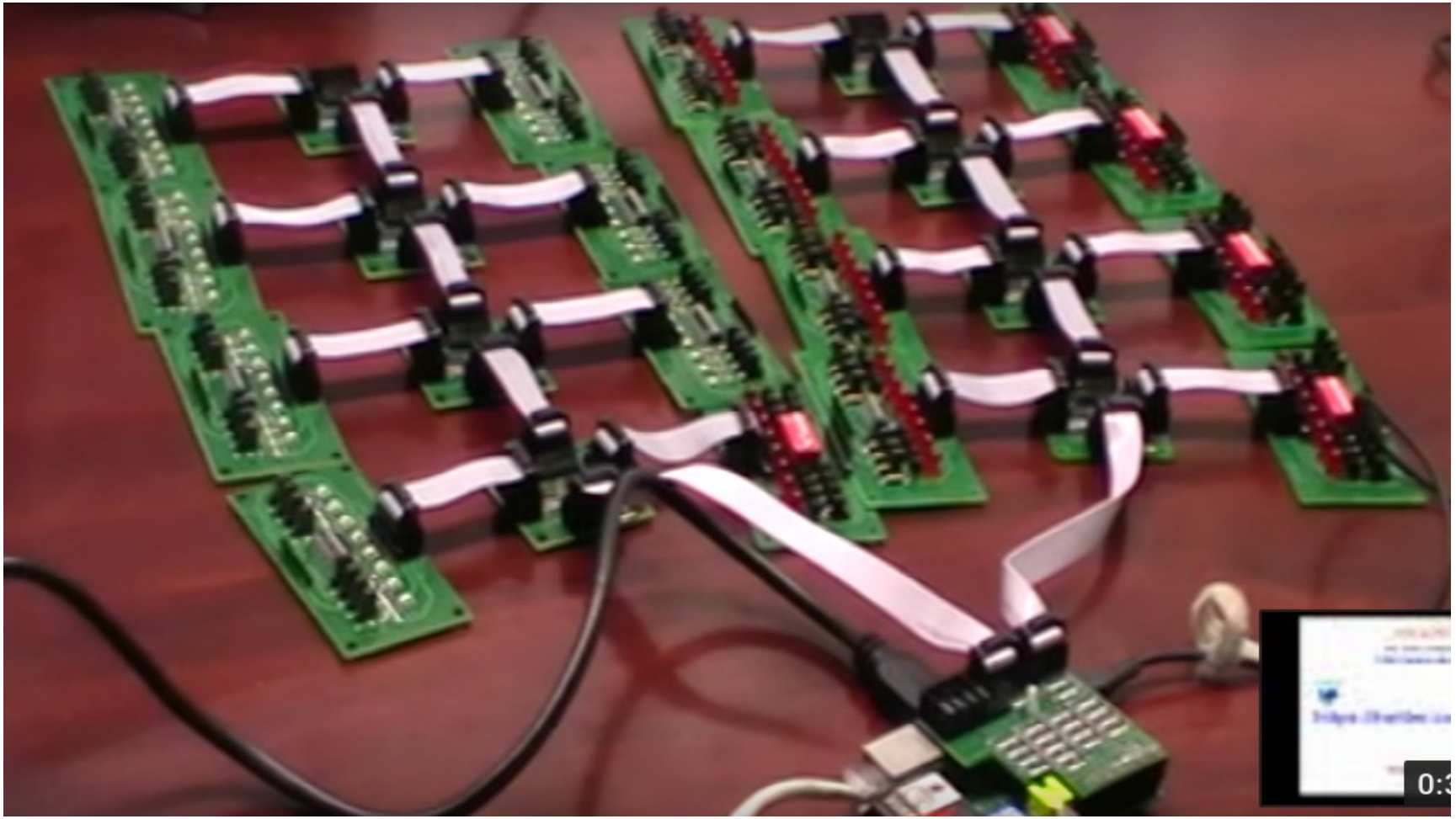
uSW 3	uSW 4	ADDR
OFF	OFF	3E
ON	OFF	3A
OFF	ON	3C
ON	ON	38

uSW 1	uSW 2	ADDR
OFF	OFF	3F
ON	OFF	3B
OFF	ON	3D
ON	ON	39

http://electronics.cat/php/common/index.php?lang=ca&page=303

Comunicacions amb Raspberry Pi

I2C



http://electronics.cat/php/common/index.php?lang=ca&page=508

Arduino i Raspberry Pi

Instal·lació de més programes

Instal·leu biblioteques complementàries per a interactuar amb Python, l'IDE d'Arduino i les biblioteques complemenàries d'electronics.cat :

```
pi@raspberrypi ~ $ sudo aptitude install python-dev python-rpi.gpio python-serial
python-netifaces python-pip

pi@raspberrypi ~ $ sudo pip install pyfirmata

pi@raspberrypi ~/downloads $ sudo aptitude install arduino

pi@raspberrypi ~/downloads $ wget http://www.electronics.cat/doc/arduinoLib/ecat.zip
```

Descomprimiu ecat.zip i moveu la carpeta ecat generada a la carpeta de biblioteques d'Arduino

```
pi@raspberrypi ~/downloads $ sudo mv ecat /usr/share/arduino/libraries
```

Descarregueu el paquet de l'Scratch For Arduino a la carpeta /home/pi/downloads i instal·leu-lo

```
pi@raspberrypi ~ $ cd downloads/
pi@raspberrypi ~/downloads $ sudo dpkg -i S4A15_RP.deb
```

Comunicacions amb Raspberry Pi

Configuració de l'accés sèrie

Instal·lació del minicom, programa de comunicacions amb el port sèrie :

```
pi@raspberrypi ~ $ sudo aptitude install minicom
```

Assignació de permisos a l'usuari normal (més enllà del primari o root) :

```
pi@raspberrypi ~ $ sudo usermod -a -G dialout pi
```

(Cal que sortiu i torneu a entrar a la sessió per a que els permisos siguin efectius)

Execució del minicom pel terminal :

minicom -b 9600 -o -D /dev/ttyAMA0

minicom -b 9600 -o -D /dev/ttyUSB0

Per a sortir del minicom :

Prémer les tecles 'Ctrl' i 'A' simulatàniament i sense deixar anar, prémer 'Z', després l'opció 'X' i seleccionar 'Yes'

Comunicacions amb Raspberry Pi

Imatge de la Raspberry Pi descarregable

Us podeu descarregar la imatge ja preparada de la Raspberry prement sobre la icona o sobre l'enllaç :

https://mega.co.nz/#!1xFVEJKK!zear_LuHkAKryys4D_wsj8_jehya_7MZ1Ath84BF--4

L'arxiu descarregat és **raspbianEcat00.img.zip** i ocupa 1,27GB. Un cop descomprimit l'arxiu **raspbianEcat00.img** ocupa 7,9GB. L'ocupació real de la imatge és de menys de 5GB. La primera opció del configurador raspi-config és per a expandir la imatge a la mida màxima de la vostra targeta SD concreta.

```
pi@raspberrypi ~ $ sudo raspi-config
```

Comunicacions amb Raspberry Pi

Càrrega del protocol Firmata a l'Arduino

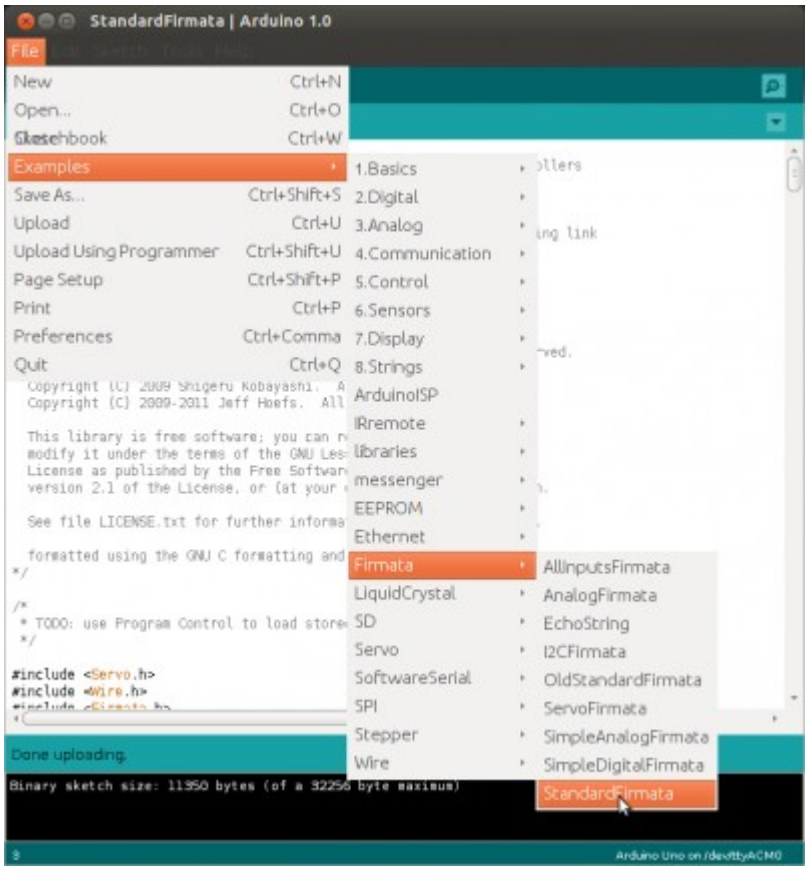
Connecteu la Raspberry Pi a la placa nano-eCat (la que conté l'Arduino Nano) mitjançant el cable USB-A a miniUSB.

A la Raspberry Pi aneu a Menú / Electrònica / Arduino IDE.

A l'Arduino IDE aneu a Fitxer / Exemples / Firmata / Standard Firmata.

Un cop l'Arduino IDE té carregat el programa Standard Firmata, premeu a la icona de la fletxa que apunta a la dreta (quan passeu el ratolí per sobre hi sortirà el text ”Puja” a la seva dreta).

Un cop hi surti el text ”Pujada enllestida” de color negre sobre fons verd a la part de sota de l'IDE, ja està preparat l'Arduino Nano per establir comunicacions seguint el protocol Firmata.



Comunicacions amb Raspberry Pi

Prova del protocol Firmata entre l'Arduino i la Raspberry Pi

A la Raspberry Pi aneu a Menú / Accessoris / LXTerminal i escriviu el següent :

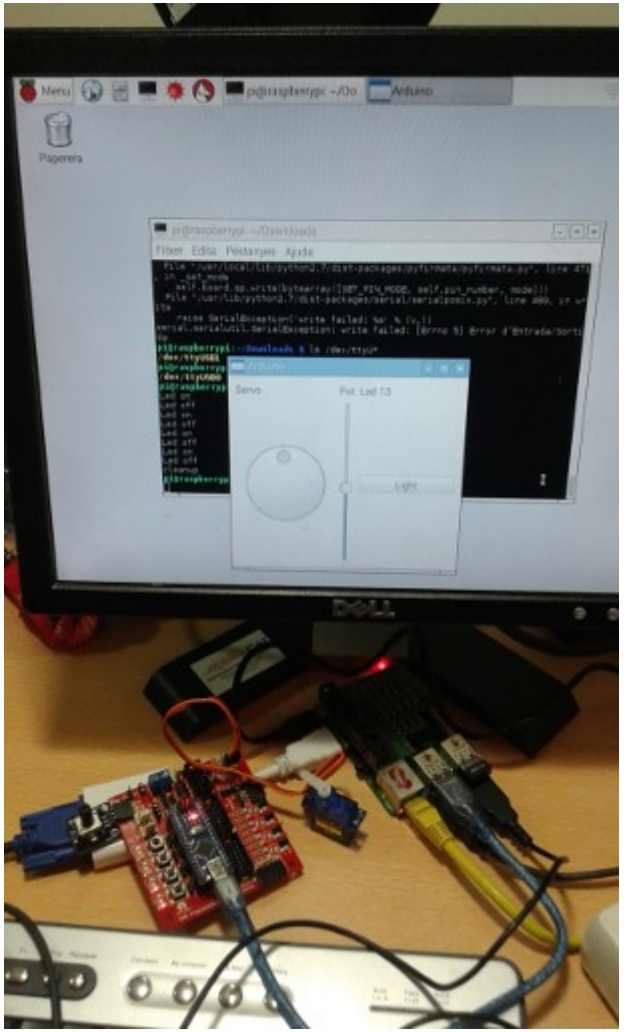
```
pi@raspberrypi ~ $ cd Documents/codis/firmata/pyQt/
pi@raspberrypi ~/Documents/codis/firmata/pyQt $ ls -ls
total 24
8 -rwxr-xr-x 1 pi pi 4122 des  7 21:10 exFirmata03P1.py
4 -rwxr-xr-x 1 pi pi 3970 des  7 20:57 exFirmata03P2.py
4 -rw-r----- 1 pi pi 3255 abr  9 13:31 led_uSw.ui
4 -rw-r--r-- 1 pi pi 3954 des  5 16:25 ui_led_uSw.py
4 -rw-r--r-- 1 pi pi 3058 abr  9 13:33 ui_led_uSw.pyc
pi@raspberrypi ~/Documents/codis/firmata/pyQt $ ./exFirmata03P2.py
```

Us sortirà la finestra de la dreta amb diverses caselles de selecció. Seleccioneu i deseleccioneu la casella corresponent a Led Bit 7 i observeu com canvia d'estat (il·luminat / apagat) el led integrat a l'Arduino.

http://binefa.cat/blog/?p=109

Comunicacions amb Raspberry Pi

Raspberry Pi amb pyfirmata i Arduino amb firmata estàndard



Codi per a la Raspberry Pi (qtgui03.py)

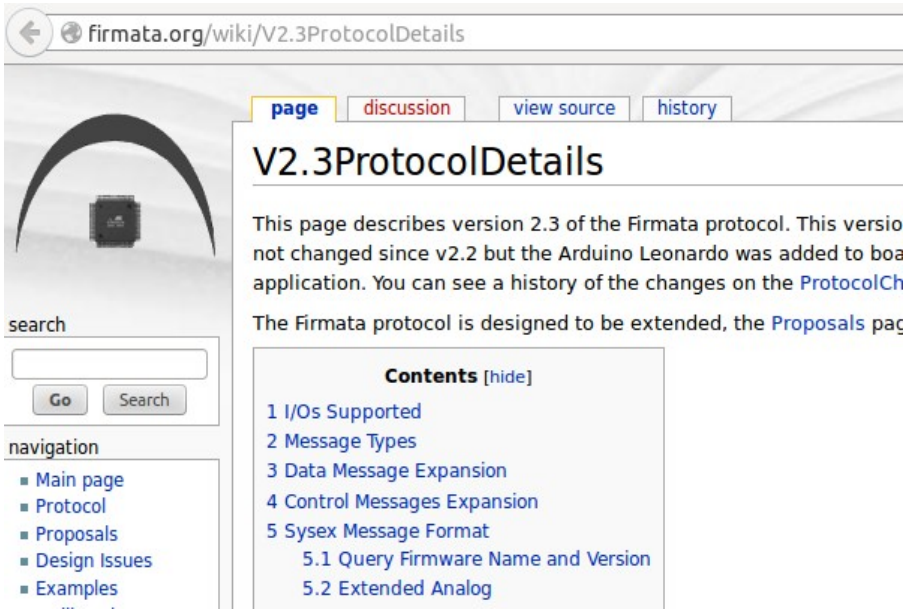
Cal tenir instal·lada la biblioteca pyfirmata:

```
sudo pip install pyfirmata
```

Més informació : http://binefa.cat/blog/?p=102
Vídeo: https://youtu.be/2kSSPvfXqug

Comunicacions amb Raspberry Pi

Protocol Firmata



http://firmata.org/wiki/V2.3ProtocolDetails

Vídeo de com funciona firmata : https://youtu.be/l4jFQIoY9ZY

Comunicacions amb Raspberry Pi

Ús de pyFirmata

Usage

Basic usage:

```
>>> from pyfirmata import Arduino, util
>>> board = Arduino('/dev/tty.usbserial-A6008rIF')
>>> board.digital[13].write(1)
```

To use analog ports, it is probably handy to start an iterator thread. Otherwise the board will keep sending data to your serial, until it overflows:

```
>>> it = util.Iterator(board)
>>> it.start()
>>> board.analog[0].enable_reporting()
>>> board.analog[0].read()
0.661440304938
```

`:` . Eg. `a:0:i` for analog 0 as input or `d:3:p` for digital pin 3 as pwm.:

```
>>> analog_0 = board.get_pin('a:0:i')
>>> analog_0.read()
0.661440304938
>>> pin3 = board.get_pin('d:3:p')
>>> pin3.write(0.6)
```

https://github.com/tino/pyFirmata

Comunicacions amb Raspberry Pi

Ús del minicom

Aprofitant que està carregat l'Arduino amb un programa que empra comunicacions (trametent un 1 s'il·lumina un led, trametent un 2 s'apaga i movent el microruptor l'Arduino tramet a la Raspberry Pi una H o una L), farem ús del programa minicom per a establir les comunicacions.

És important assegurar-se de que cap altre programa està fent ús del canal sèrie. Ni l'Arduino IDE ni l'ArduBlock.

```
pi@raspberrypi ~ $ minicom -b 9600 -o -D /dev/ttyUSB0
```

```
Welcome to minicom 2.6.1

OPTIONS: I18n
Compiled on Apr 28 2012, 19:24:31.
Port /dev/ttyUSB0

Press CTRL-A Z for help on special keys

L
H
```

Prement 1 s'il·lumina un led, trametent un 2 s'apaga (quan escrivim no hi surt res pel minicom perque l'Arduino no fa eco del que li trametem) i movent el microruptor l'Arduino tramet a la Raspberry Pi una H o una L. Per sortir-ne : Ctrl A + Z, X i Yes

Comunicacions amb Raspberry Pi

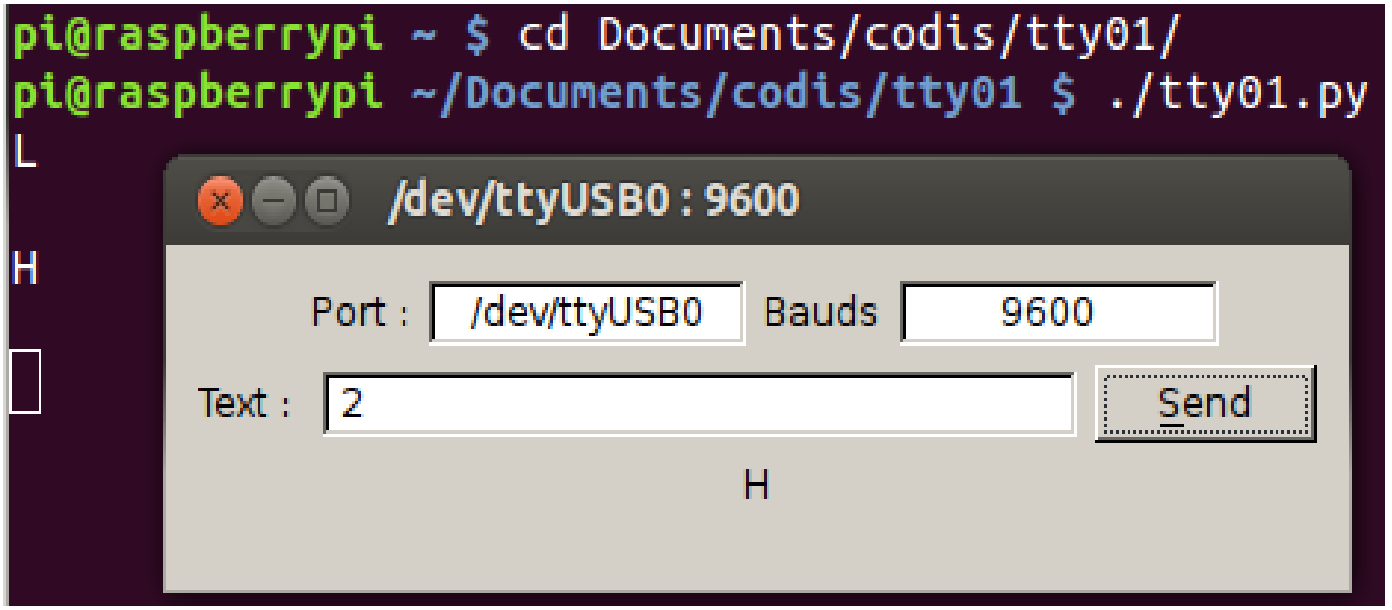
Ús del port sèrie des de Python

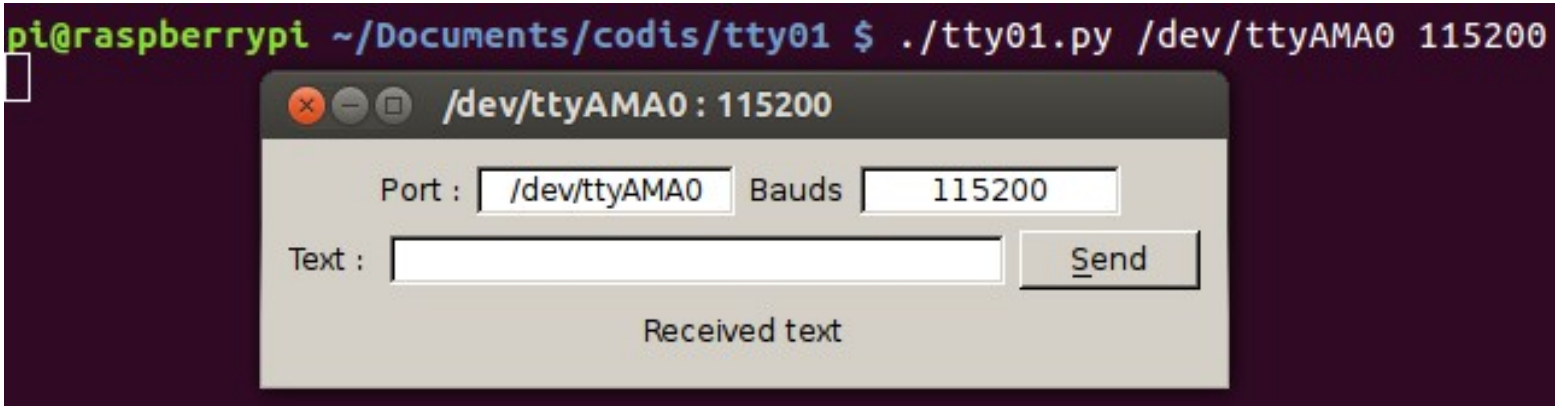
```
1  #!/usr/bin/python
2
3  import serial
4
5  print "Python serial port sample"
6
7  rpiSerialPort = '/dev/ttyAMA0'
8  ser = serial.Serial(rpiSerialPort,115200)
9
10 ser.write("\n\rFrom Raspberry Pi\n\r")
11 ser.close()
12
```

https://dl.dropboxusercontent.com/u/65254823/oshw20130912/serial00.py

Comunicacions amb Raspberry Pi

Ús d'un programa amb interfície gràfica





http://www.binefa.cat/php/RPi/tty01.py.zip

Comunicacions amb Raspberry Pi

Codi d'Arduino

```
/*
serialWrRd01.ino

http://www.binefa.cat/php/arduino/serialWrRd01.zip
*/

void setup() {
// Velocitat de la comunicacio en bauds
  Serial.begin(9600);
}

void loop() {
  byte byteLlegit;
   /*  Si hi ha quelcom a llegir: */
  if (Serial.available()) {
    /* llegeix el byte mes recent */
    byteLlegit = Serial.read();
    /* Retorna el valor llegit. Es a dir, fa eco.*/
    Serial.write(byteLlegit);
  }
}
```

http://www.binefa.cat/php/arduino/serialWrRd01.zip

```
/*
serialWrRd02.ino
http://www.binefa.cat/php/arduino/serialWrRd02.zip
*/
#include <ecat.h>
Ecat ecat;

void setup() {
  // Els 4 bits (nibble) baixos de P2 son d'entrada
  ecat.setupNibbleMode(NIBBLE_L_P2,INPUT);
  // Els 4 bits (nibble) alts de P2 son de sortida
  ecat.setupNibbleMode(NIBBLE_H_P2,OUTPUT);
  // Velocitat de la comunicacio en bauds
  Serial.begin(9600);
}

void loop() {
  byte byteLlegit;
   /*  Si hi ha quelcom a llegir: */
  if (Serial.available()) {
    /* llegeix el byte mes recent */
    byteLlegit = Serial.read();
    /* Retorna el valor llegit. Es a dir, fa eco.*/
    Serial.write(byteLlegit);
    if(byteLlegit == '1'){
      digitalWrite(ecat.nPinP2B7,HIGH);
    }
    if(byteLlegit == '2'){
      digitalWrite(ecat.nPinP2B7,LOW);
    }
  }
}
```

http://www.binefa.cat/php/arduino/serialWrRd03.zip

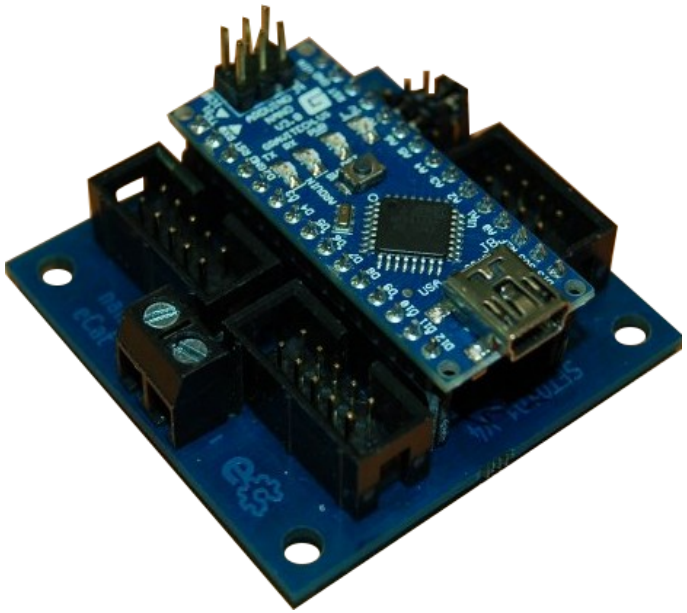
```
/*
serialWrRd03.ino
http://www.binefa.cat/php/arduino/serialWrRd03.zip
*/
#include <ecat.h>
Ecat ecat;

void setup() {
  ecat.setupNibbleMode(NIBBLE_L_P2,INPUT);
  ecat.setupNibbleMode(NIBBLE_H_P2,OUTPUT);
  Serial.begin(9600);
}

void vIterpretaByteLlegit(byte by){
    if(by == '1')
      digitalWrite(ecat.nPinP2B7,HIGH);
    if(by == '2')
      digitalWrite(ecat.nPinP2B7,LOW);
}

bool bCanviP2B0(){
  static bool bNivellP2B0 = digitalRead(ecat.nPinP2B0);

  if(bNivellP2B0 != digitalRead(ecat.nPinP2B0)){
    bNivellP2B0 = digitalRead(ecat.nPinP2B0);
    return true;
  }
  return false;
}

void vTrametValor(byte byQuinPin){
  if(digitalRead(ecat.nPinP2B0))
    Serial.write('H');
  else
    Serial.write('L');
}

void loop() {
  byte byteLlegit;

  if (Serial.available()) {
    byteLlegit = Serial.read();
    Serial.write(byteLlegit);
    vIterpretaByteLlegit(byteLlegit);
  }
  if( bCanviP2B0() ){
    delay(100); // Per a evitar rebots
    vTrametValor(ecat.nPinP2B0);
  }
}
```

Maquinari lliure amb Arduino Nano

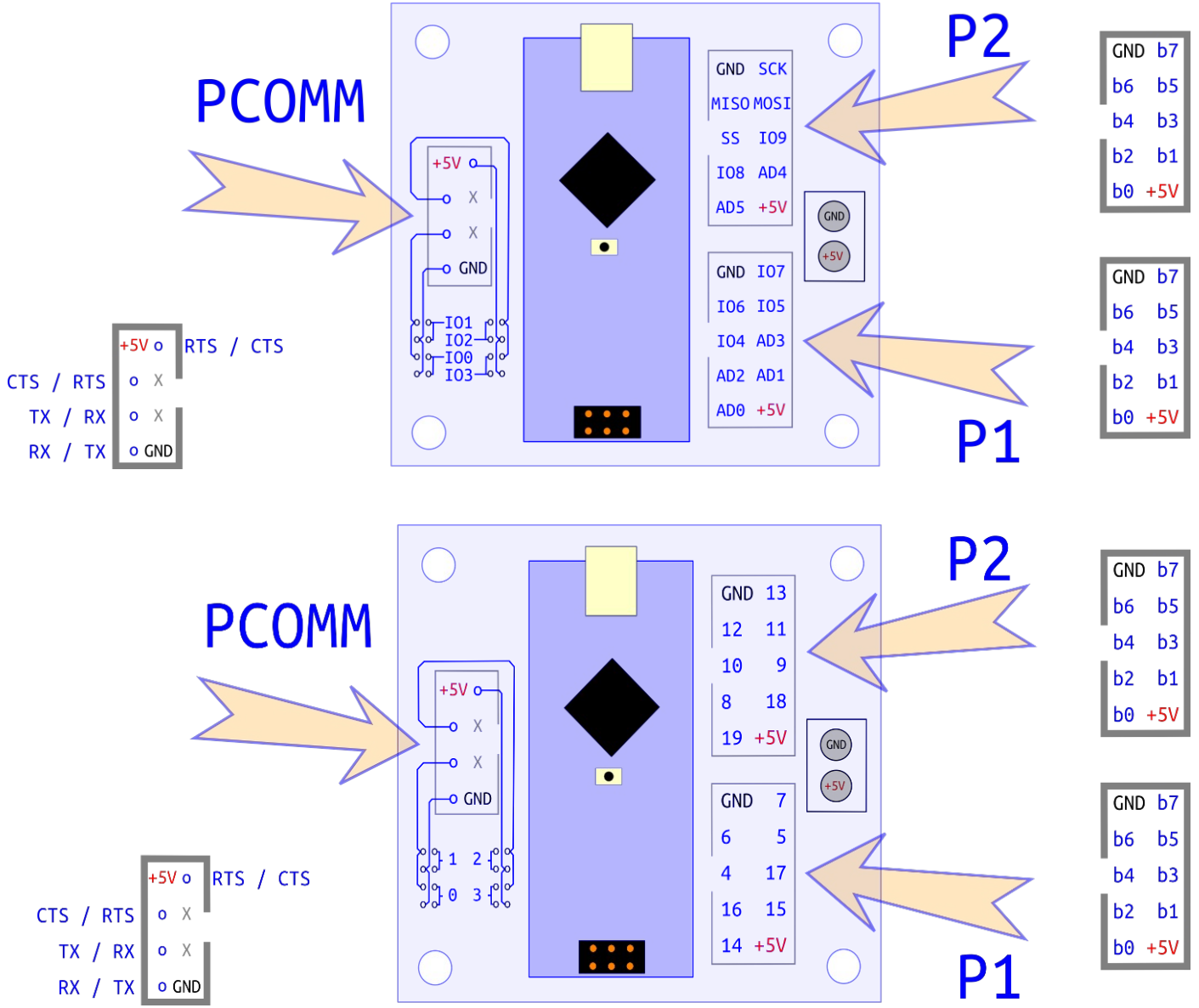
Plaques amb sistema de bus lliure eCat

nano-eCat

Placa 0101 v4

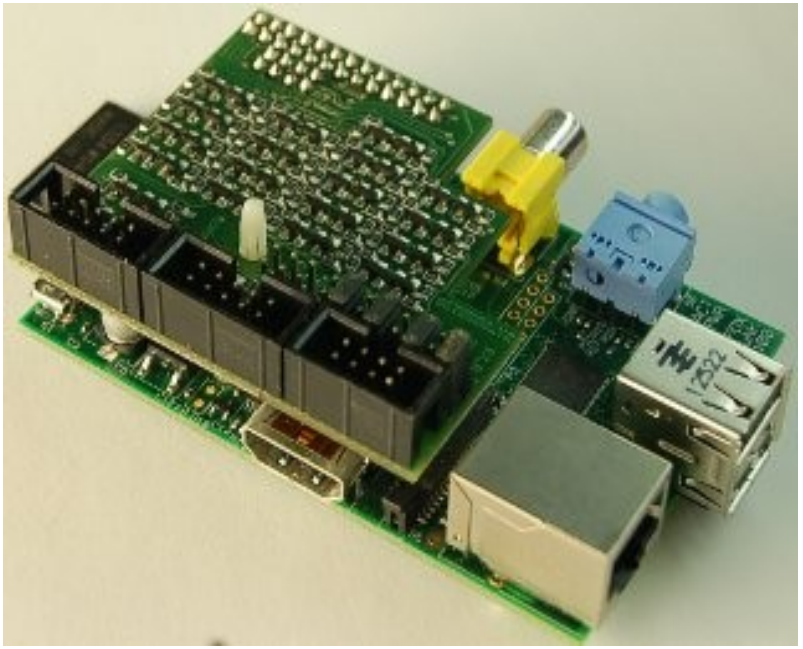


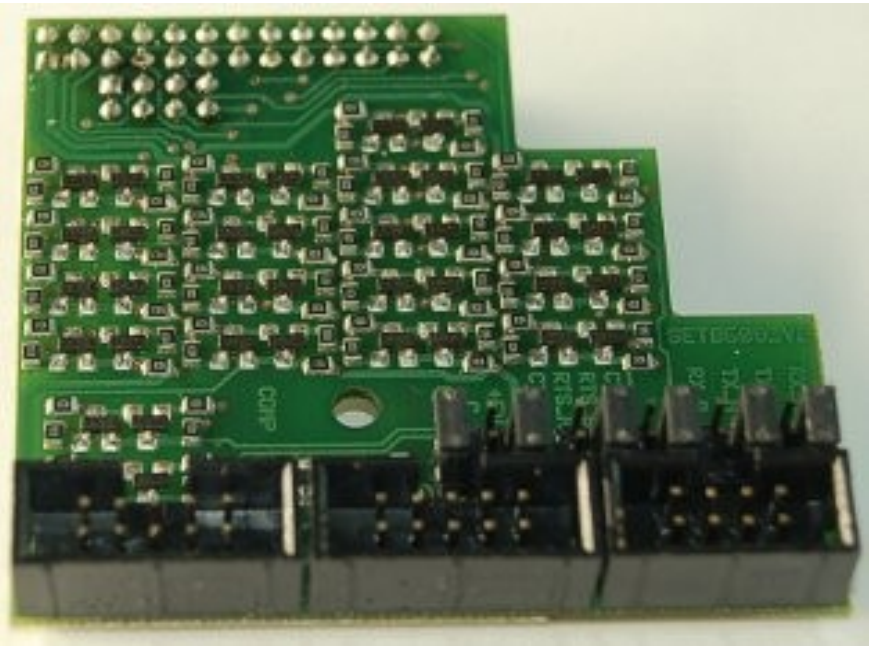
Placa convertidora de l'Arduino Nano al sistema de bus lliure eCat System



Maquinari

3Bpi - Pi de les Tres Branques
Placa traductora de nivells per a Raspberry Pi
Placa set05_08_v2





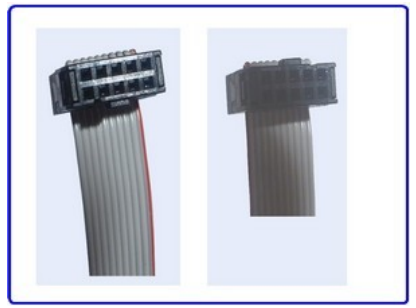
http://electronics.cat/php/common/index.php?lang=ca&page=508

Arduino i Raspberry Pi

BUS : Sistema digital que transfereix dades entre dispositius electrònics

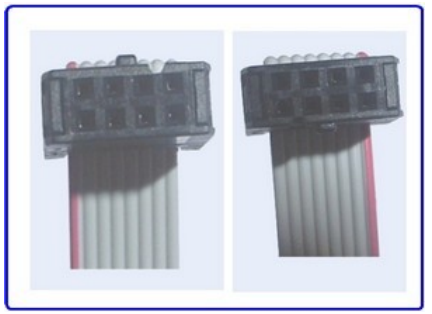
GND	b7
b6	b5
b4	b3
b2	b1
b0	+5V

Cable pla
5x2 a 5x2
(0104A)



	+5V o	CTS / RTS
CTS / RTS	o X	
TX / RX	o X	
RX / TX	o GND	

Cable pla
4x2 a 4x2
(0311)

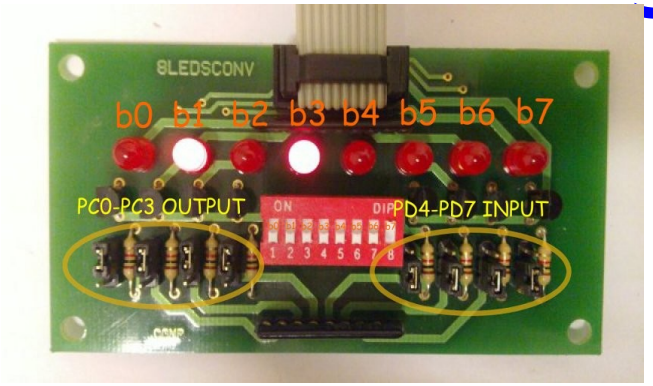


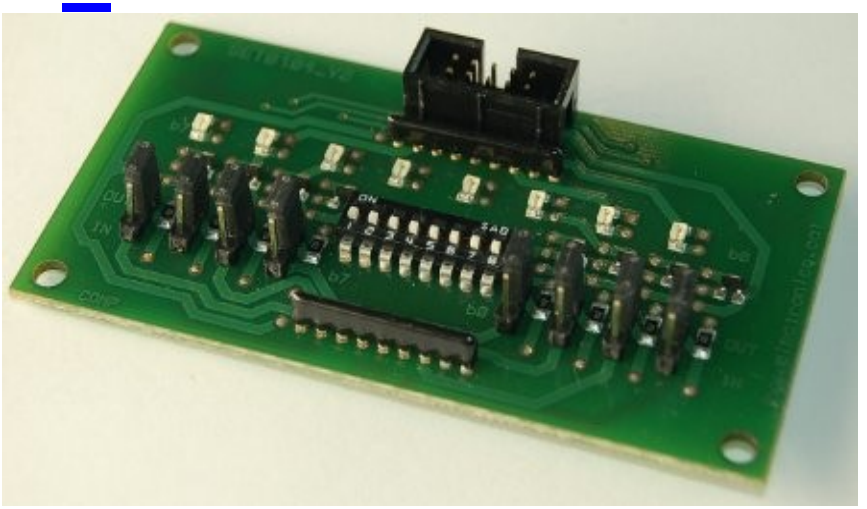
eCat System : Dues tipologies de BUS, dades (8 bits) i comunicacions

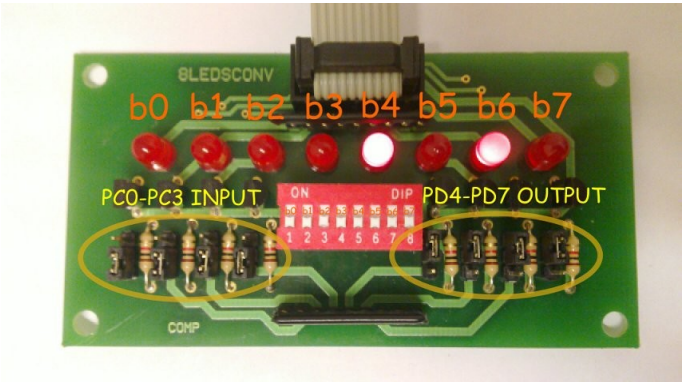
http://electronics.cat/php/common/index.php?lang=ca&page=044

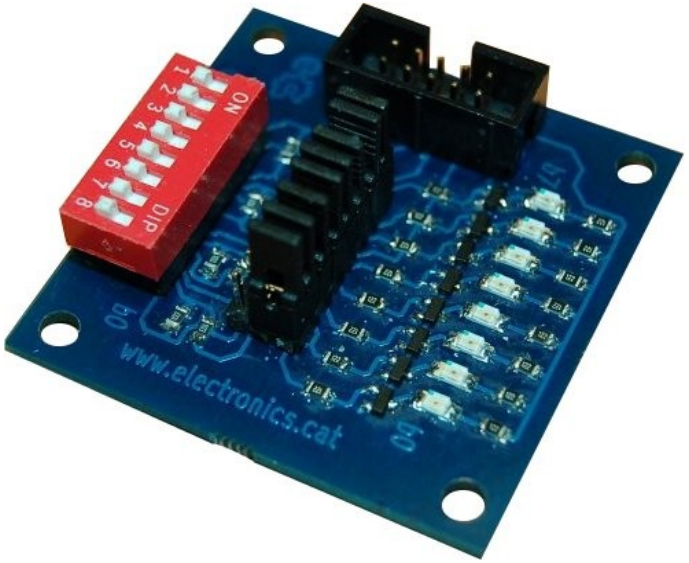
Arduino i Raspberry Pi

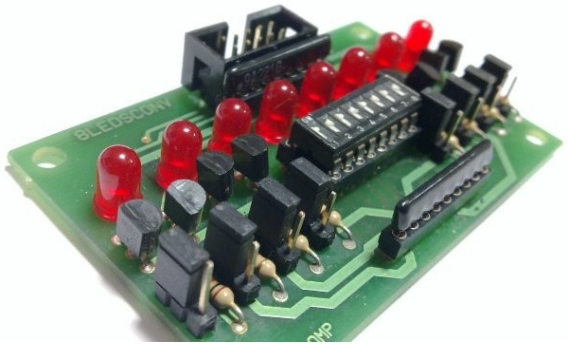
Leds i microrruptors - Placa set01_04











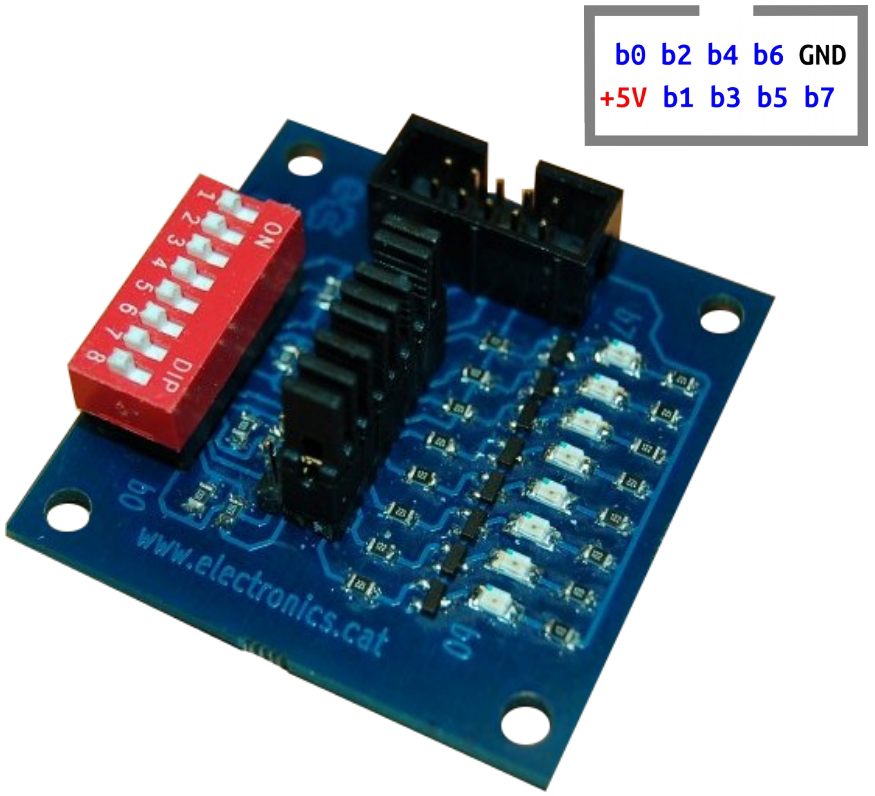
http://electronics.cat/php/common/index.php?lang=ca&page=104

Arduino i Raspberry Pi

Plaques amb sistema de bus lliure eCat

uSW – LEDs

01_04 v3



Placa configurable d'entrades / sortides.

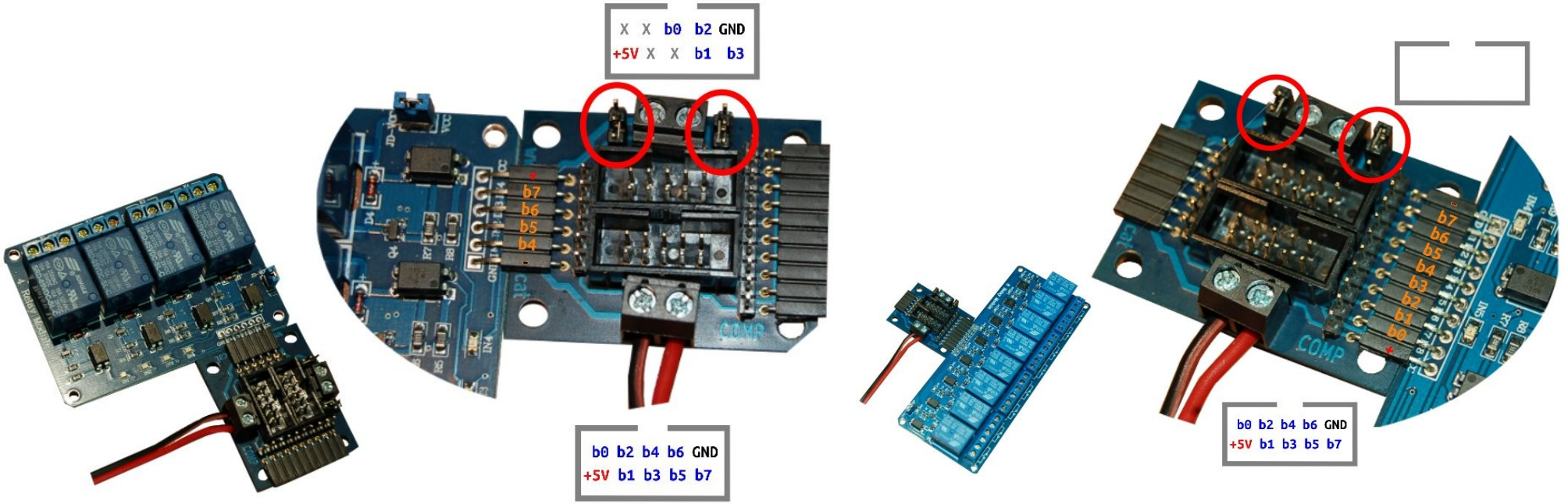
La configuració de cada bit es fa mitjançant un pont (jumper).

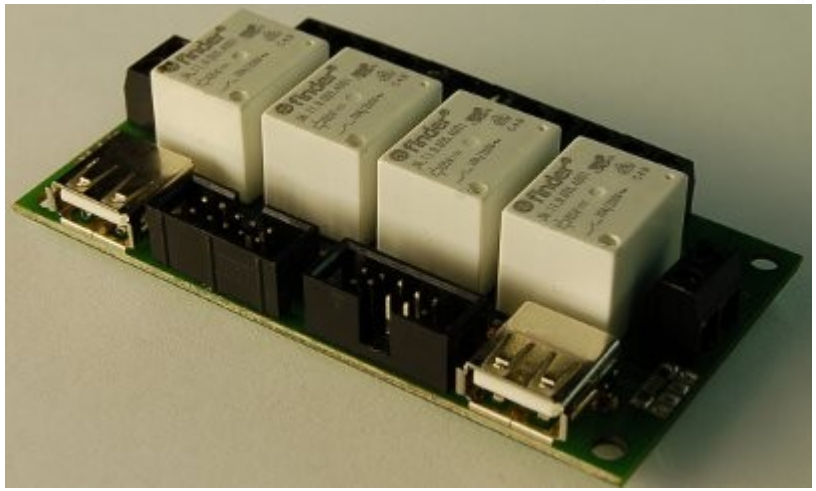
Les entrades es fan per microrruptor (ON : zero lògic, OFF : u lògic).

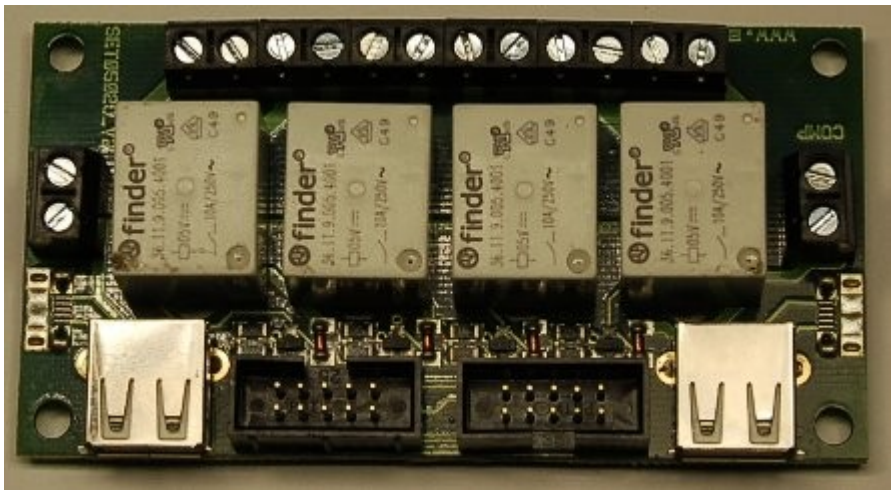
Les sortides es visualitzen mitjançant leds (apagat : zero lògic, encès : u lògic).

Maquinari

4 Relés - Placa set05_02

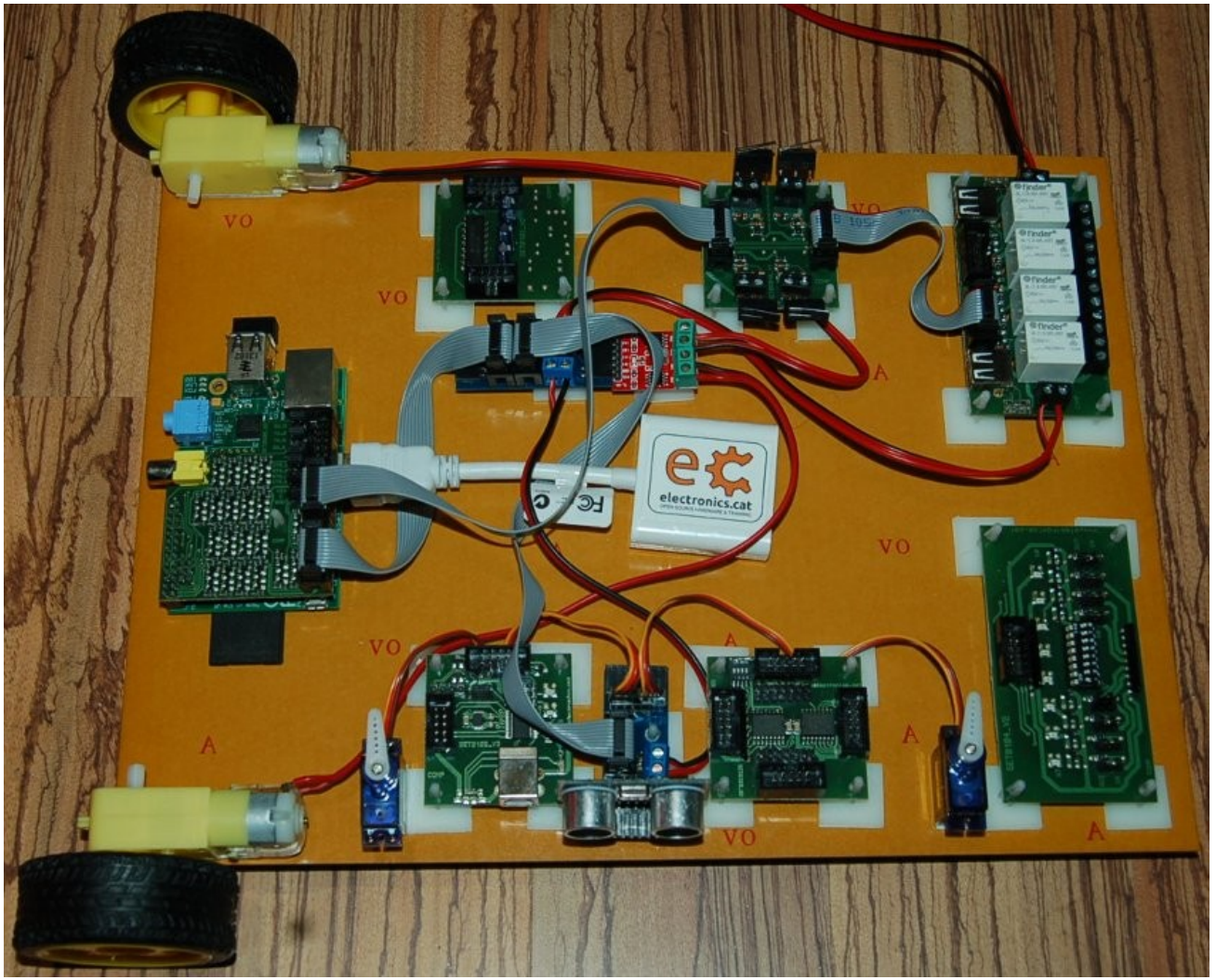






http://electronics.cat/php/common/index.php?lang=ca&page=502

Arduino i Raspberry Pi



Conjunt d'una Raspberry Pi amb plaques de http://www.electronics.cat/

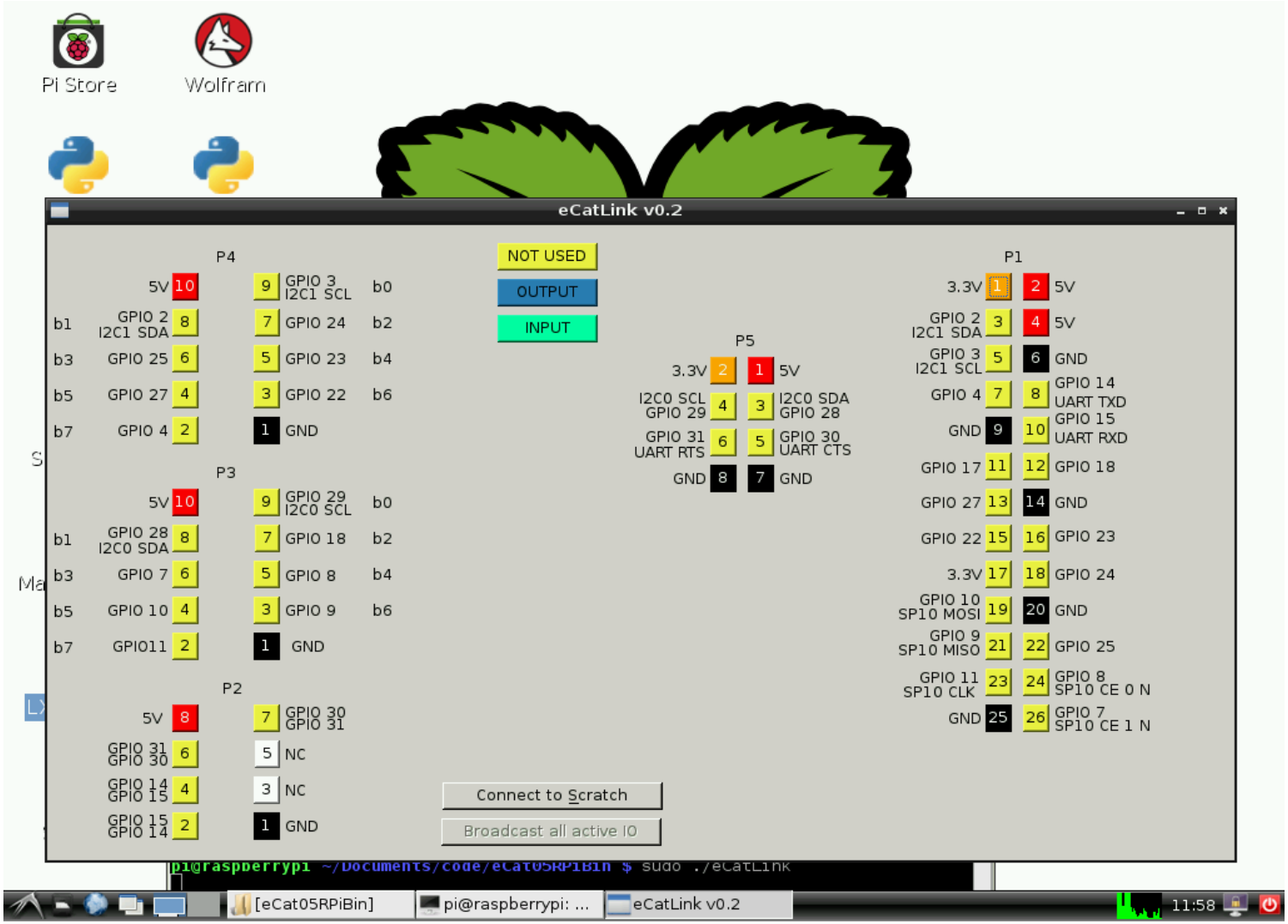
Accés a la GPIO emprant Qt

Code less.
Create more.
Deploy everywhere.

http://qt-project.org/

Accés a la GPIO emprant Qt

eCatLink



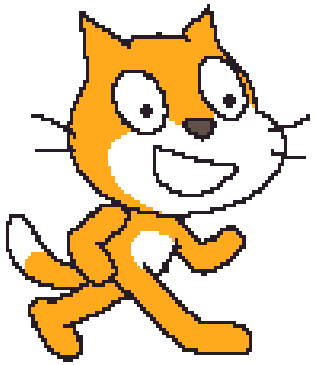
http://www.binefa.cat/php/raspberryPi/qt/eCat05b.tar.gz

Accés a la GPIO emprant Qt

eCatLink

http://www.binefa.cat/php/raspberryPi/qt/eCat05b.tar.gz

SCRATCH



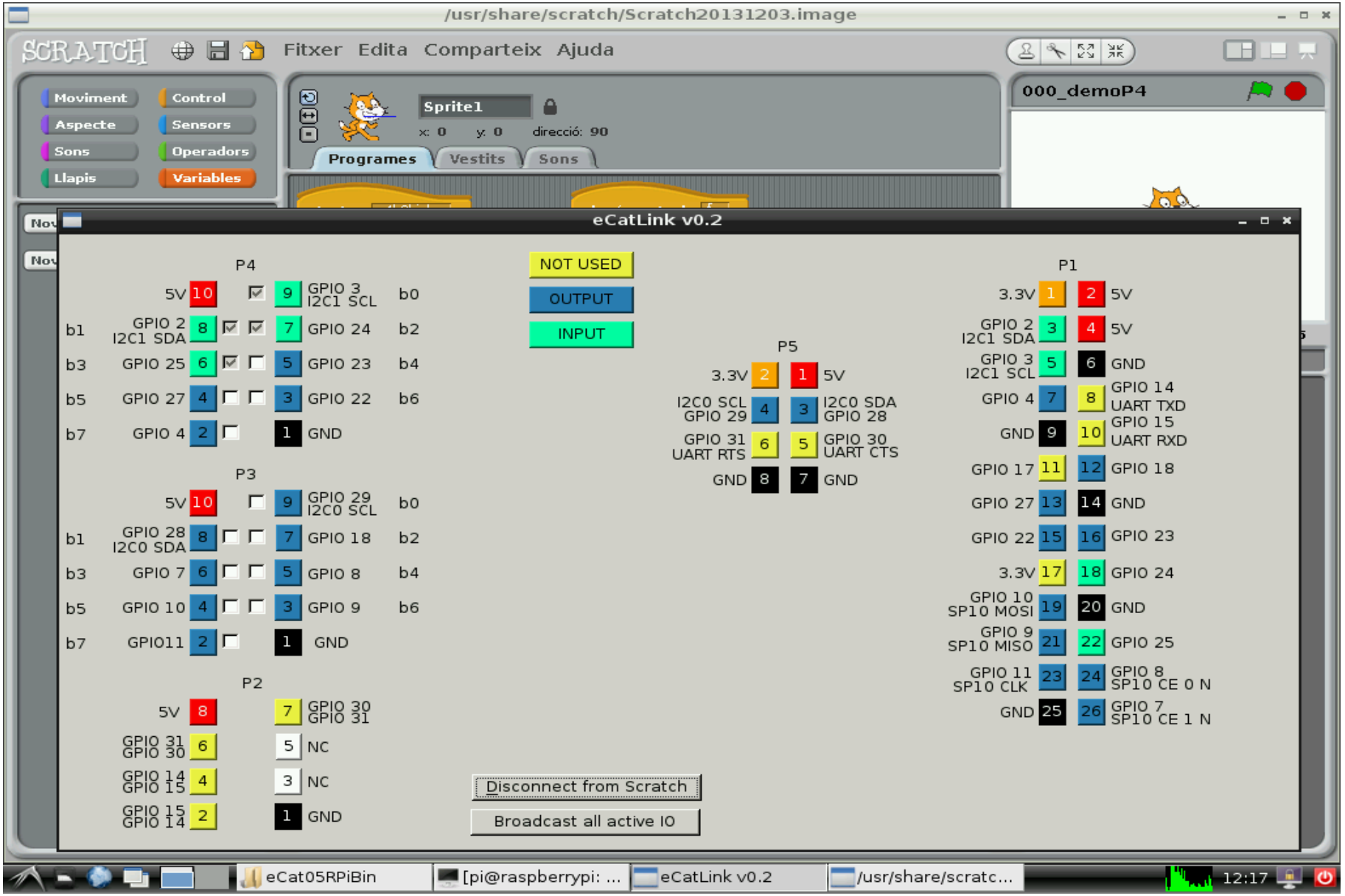
+

Qt

http://oshw.binefa.net/?p=4

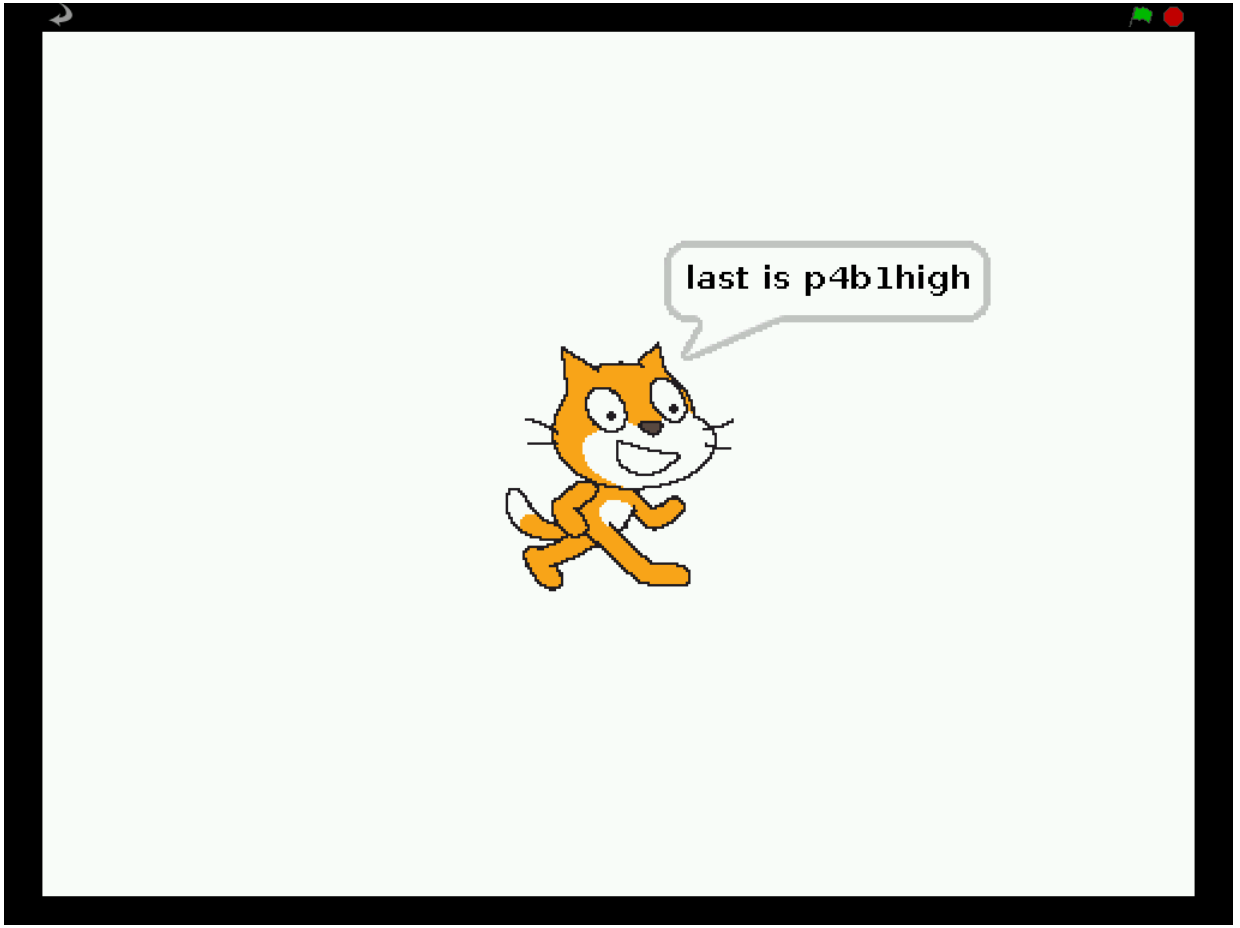
Maquinari

Fent ús de P4 - eCatLink



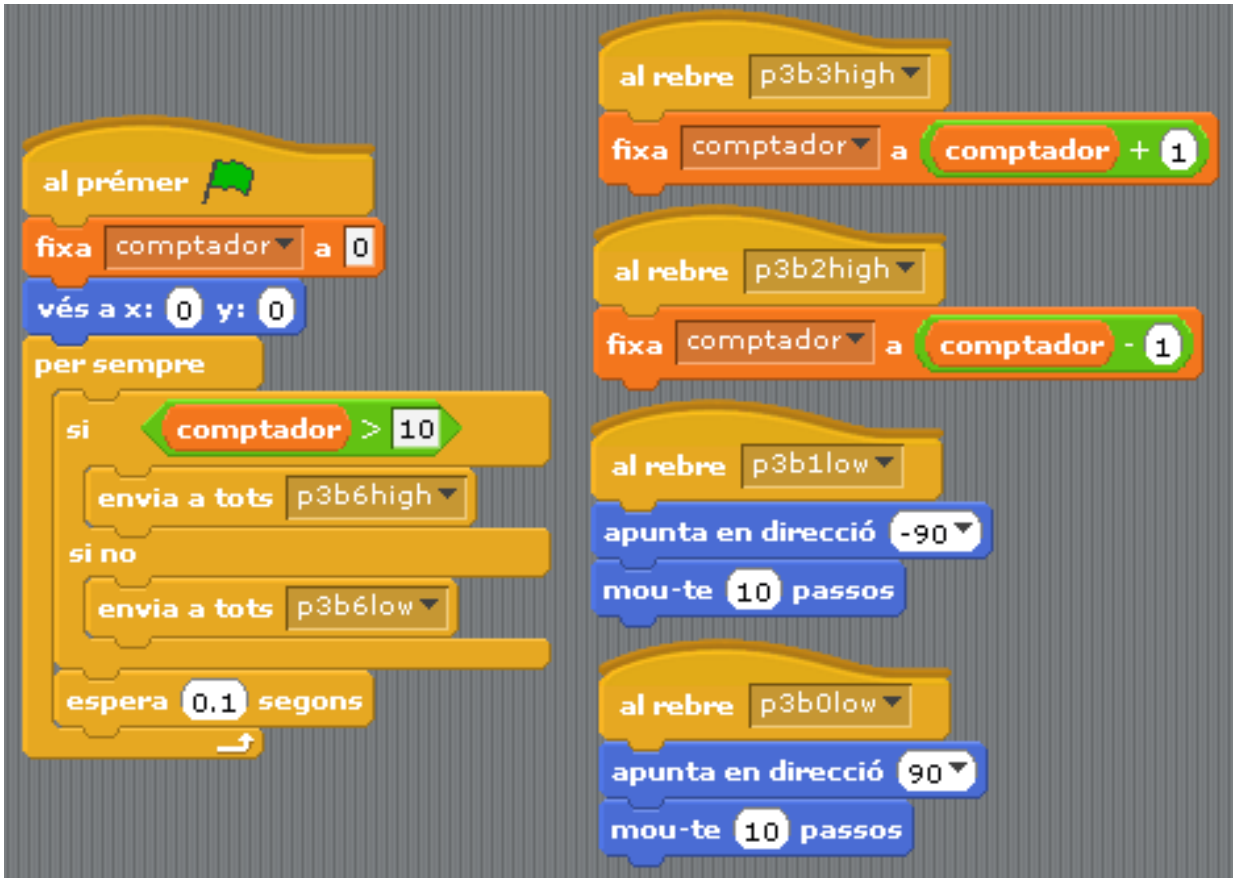
Maquinari

eCatLink + Scratch



Maquinari

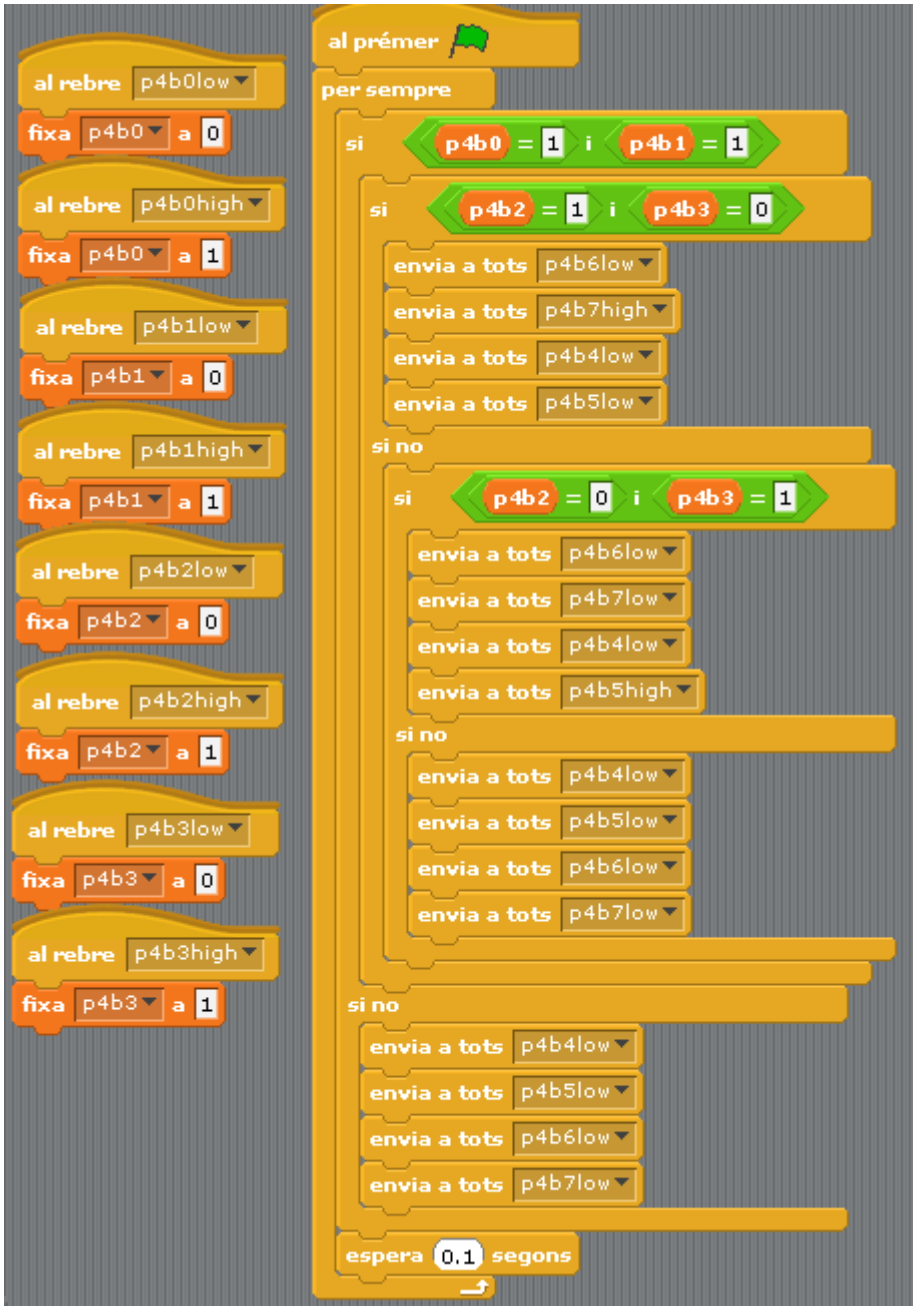
eCatLink + Scratch



http://www.binefa.cat/php/raspberryPi/scratch/002_cmpt.sb

Maquinari

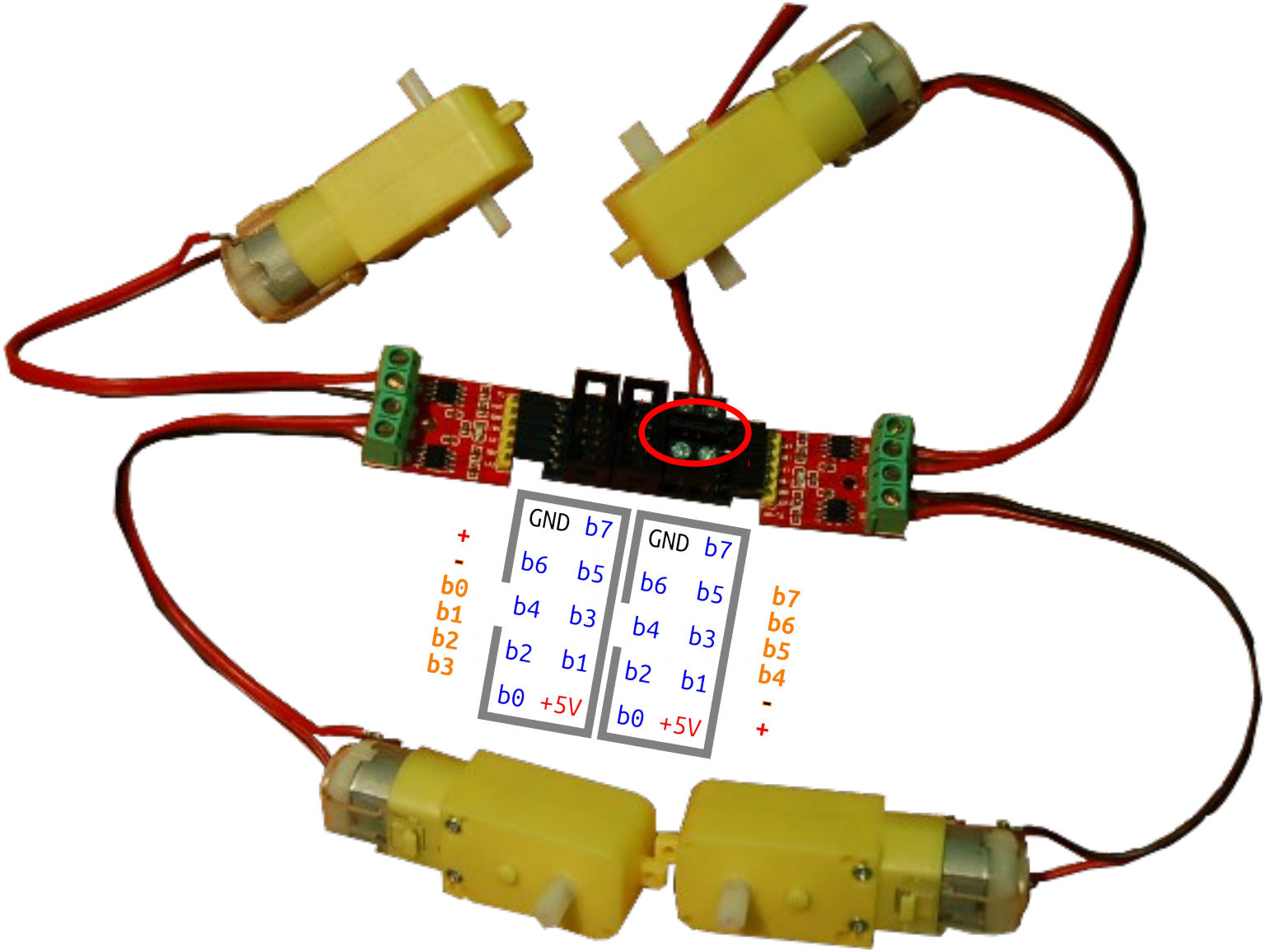
eCatLink + Scratch



http://www.binefa.cat/php/raspberryPi/scratch/003_motorsSensors.sb

Maquinari

Motors emprant mòdul en H



http://electronics.cat/php/common/index.php?lang=ca&page=520

Accés a la GPIO emprant Python

Instal·lació del mòdul RPi.GPIO

$sudo apt-get update
$sudo apt-get install python-dev
$sudo apt-get install python-rpi.gpio
$sudo aptitude install python-serial
$sudo aptitude install python-netifaces

$sudo usermod -a -G dialout pi

http://code.google.com/p/raspberry-gpio-python/wiki/BasicUsage

Rpi.GPIO llegint entrades de P3

```
1  #!/usr/bin/env python
2
3  from time import sleep
4  import os
5  import RPi.GPIO as GPIO
6
7  GPIO.setmode(GPIO.BCM)
8  GPIO.setup(29, GPIO.IN)
9  GPIO.setup(28, GPIO.IN)
10 GPIO.setup(18, GPIO.IN)
11 GPIO.setup(7, GPIO.IN)
12
13 while True:
14         if ( GPIO.input(29) == False ):
15                 print "GPIO 29 pressed"
16         if ( GPIO.input(28) == False ):
17                 print "GPIO 28 pressed"
18         if ( GPIO.input(18) == False ):
19                 print "GPIO 18 pressed"
20         if ( GPIO.input(7) == False ):
21                 print "GPIO 7 pressed"
22         sleep(0.1);
23
```

http://www.binefa.cat/php/raspberryPi/python/py00.py

Rpi.GPIO escrivint sortides a P3


```
1  #!/usr/bin/env python
2
3  from time import sleep
4  import os
5  import RPi.GPIO as GPIO
6
7  GPIO.setwarnings(False)
8
9  GPIO.setmode(GPIO.BCM)
10 GPIO.setup(8, GPIO.OUT)
11 GPIO.setup(10, GPIO.OUT)
12 GPIO.setup(9, GPIO.OUT)
13 GPIO.setup(11, GPIO.OUT)
14
15 GPIO.output(8,0)
16 GPIO.output(10,0)
17 GPIO.output(9,0)
18 GPIO.output(11,0)
19
```

```
19
20 while True:
21         GPIO.output(8,1)
22         sleep(0.5);
23         GPIO.output(8,0)
24         sleep(0.5);
25         GPIO.output(10,1)
26         sleep(0.5);
27         GPIO.output(10,0)
28         sleep(0.5);
29         GPIO.output(9,1)
30         sleep(0.5);
31         GPIO.output(9,0)
32         sleep(0.5);
33         GPIO.output(11,1)
34         sleep(0.5);
35         GPIO.output(11,0)
36         sleep(0.5);
```

http://www.binefa.cat/php/raspberryPi/python/py01.py

Control d'un servo – bit7 de P4

Control d'un servo – p4b2 i p4b3

```
#!/usr/bin/env python

from time import sleep
import os
import RPi.GPIO as GPIO

GPIO.setwarnings(False)

GPIO.setmode(GPIO.BCM)
GPIO.setup(24, GPIO.OUT)
GPIO.setup(25, GPIO.OUT)
p = GPIO.PWM(24,50)          # 50Hz
q = GPIO.PWM(25,50)          # 50Hz
p.start(7.5)
q.start(7.5)
try:
    while True:
        p.ChangeDutyCycle(7.5)  # Neutral
        q.ChangeDutyCycle(7.5)  # Neutral
        sleep(1)
#       p.ChangeDutyCycle(12.5) # 180 graus
        p.ChangeDutyCycle(10)   # 180 graus
        q.ChangeDutyCycle(10)   # 180 graus
        sleep(1)
#       p.ChangeDutyCycle(2.5)  # 0 graus
        p.ChangeDutyCycle(4)    # 0 graus
        q.ChangeDutyCycle(4)    # 0 graus
        sleep(1)

except KeyboardInterrupt:
    GPIO.cleanup()
```

http://www.binefa.cat/php/raspberryPi/python/pyServos.py

Adaptació de codi basat en exemples de http://www.theraspberrypiguy.com/

Lectura de sensor d'ultrasons

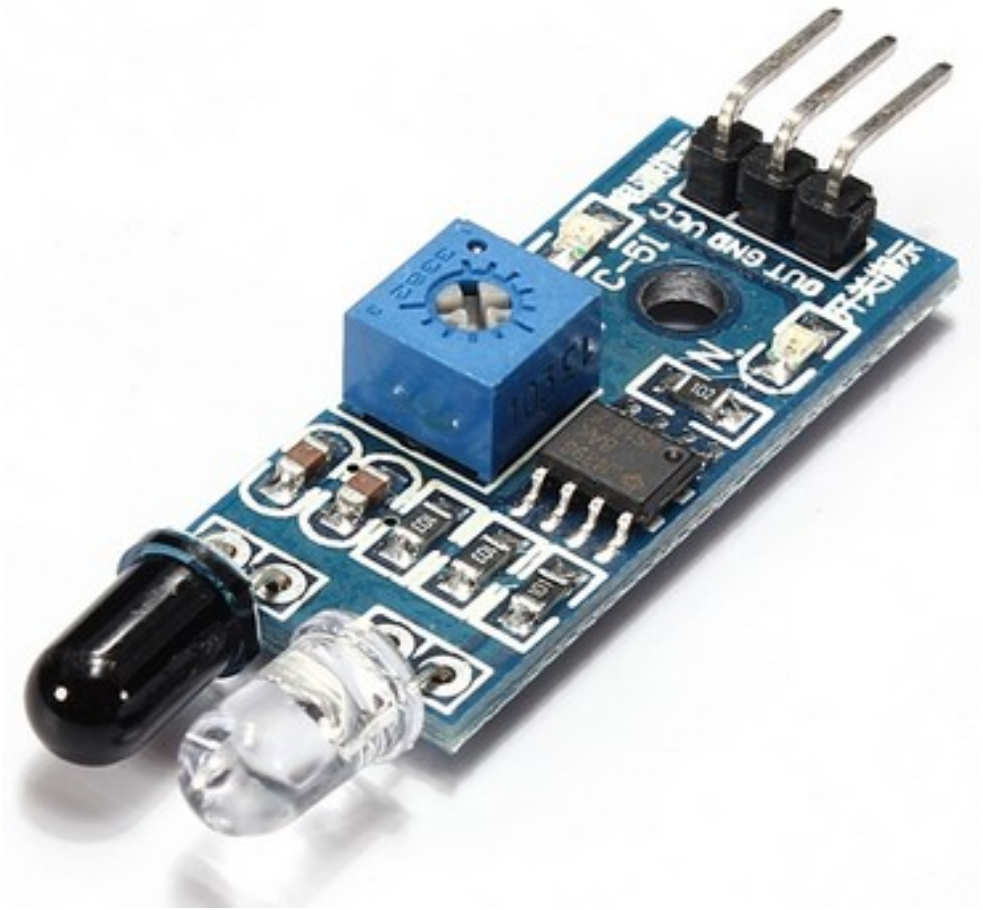
Lectura de sensor d'ultrasons

```
#!/usr/bin/python

def reading(sensor):
    import time
    import RPi.GPIO as GPIO

    GPIO.setwarnings(False)
    GPIO.setmode(GPIO.BCM)

    if sensor == 0:
        GPIO.setup(22,GPIO.OUT)
        GPIO.setup(27,GPIO.IN)
        GPIO.output(22, GPIO.LOW)
        time.sleep(0.3)

        GPIO.output(22, True)
        time.sleep(0.00001)
        GPIO.output(22, False)

        while GPIO.input(27) == 0:
          signaloff = time.time()
        while GPIO.input(27) == 1:
          signalon = time.time()

        timepassed = signalon - signaloff
        distance = timepassed * 17000

        GPIO.cleanup()
        return distance

    else:
        print "Incorrect usonic() function varible."

print reading(0)
```

http://www.binefa.cat/php/raspberryPi/python/pyUltrasons.py

Adaptació de codi basat en exemples de http://www.theraspberrypiguy.com/

Lectura de sensor infrarroig



Lectura de sensor infrarroig

```
1   #!/usr/bin/env python
2
3   from time import sleep
4   import os
5   import RPi.GPIO as GPIO
6
7   GPIO.setmode(GPIO.BCM)
8   GPIO.setup(23, GPIO.IN)
9
10  while True:
11          if ( GPIO.input(23) == True ):
12                  print "Tapat"
13          sleep(0.1);
14
```

https://dl.dropboxusercontent.com/u/65254823/oshw20140319/py05.py

Adaptació de codi basat en exemples de http://www.theraspberrypiguy.com/

Motor pas a pas

Motor pas a pas

```
#!/usr/bin/python
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)
ControlPin = [3,2,24,25]

for pin in ControlPin:
    GPIO.setup(pin,GPIO.OUT)
    GPIO.output(pin,0)

seq = [ [0,0,0,1],
    [0,0,1,1],
    [0,0,1,0],
    [0,1,1,0],
    [0,1,0,0],
    [1,1,0,0],
    [1,0,0,0],
    [1,0,0,1] ]

for i in range(512):
    for halfstep in range(8):
        for pin in range(4):
            GPIO.output(ControlPin[pin],seq[halfstep][pin])
        time.sleep(0.001)
```

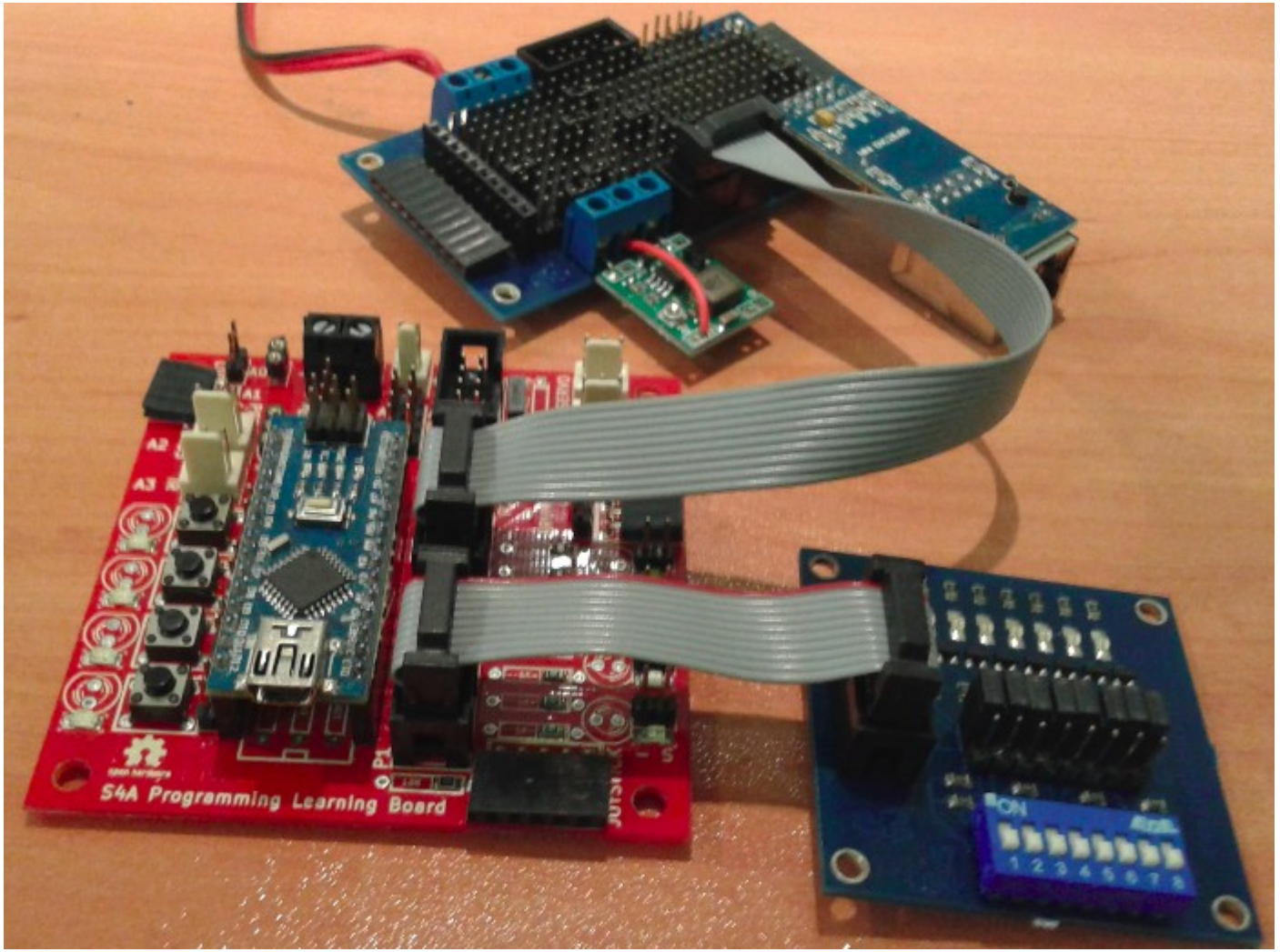
```
seq = [ [1,0,0,0],
    [1,1,0,0],
    [0,1,0,0],
    [0,1,1,0],
    [0,0,1,0],
    [0,0,1,1],
    [0,0,0,1],
    [1,0,0,1] ]
```

https://dl.dropboxusercontent.com/u/65254823/oshw20140319/py04.py
https://dl.dropboxusercontent.com/u/65254823/oshw20140319/py04b.py

Adaptació de codi basat en exemples de http://www.theraspberrypiguy.com/

Arduino i Raspberry Pi

Ethernet



Codi per a Arduino UdpServer01b

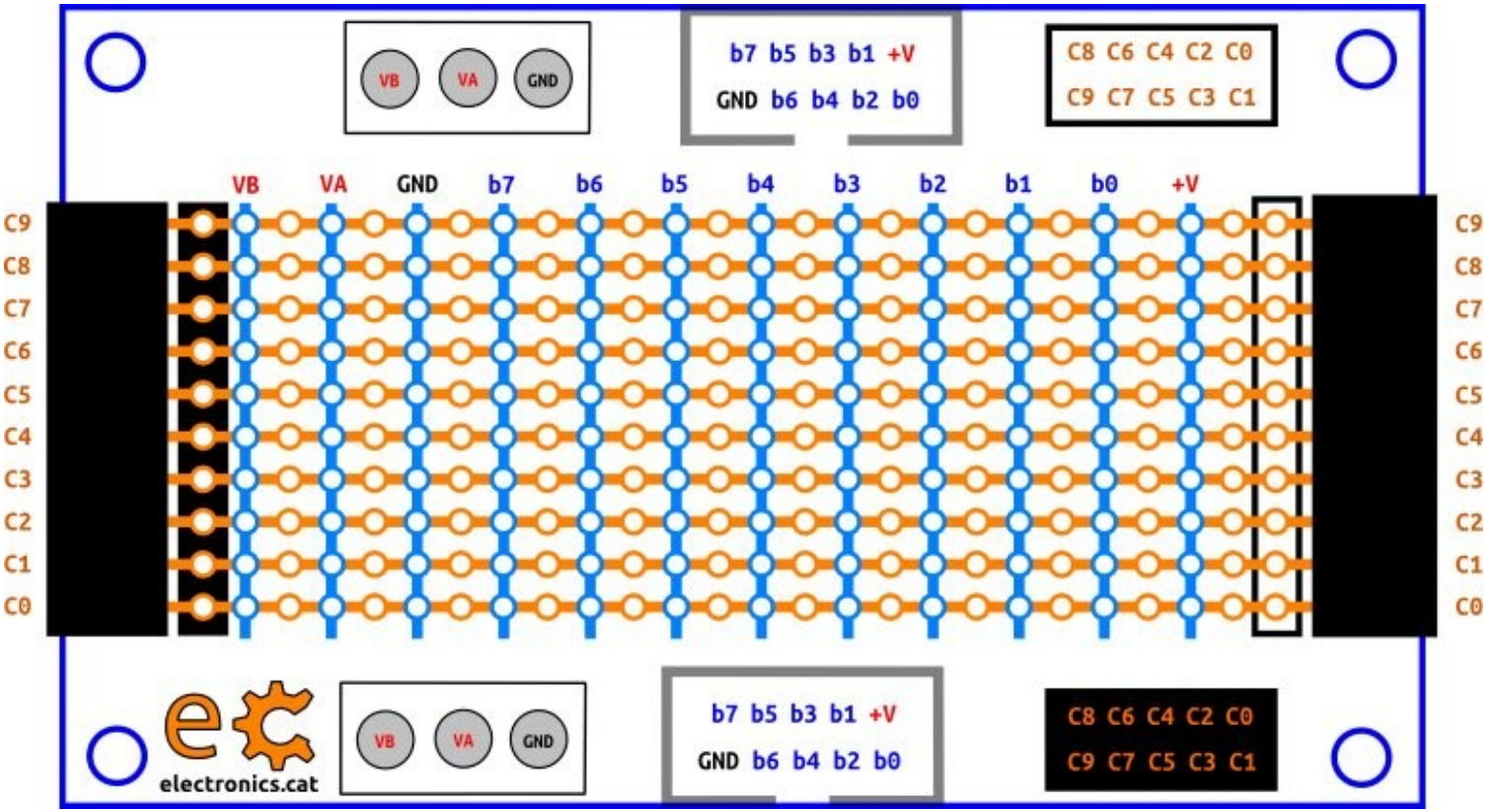
Arduino i Raspberry Pi

Ethernet – Mòdul ENC28J60

Biblioteca UIPEthernet per Arduino i ENC28J60

Arduino i Raspberry Pi

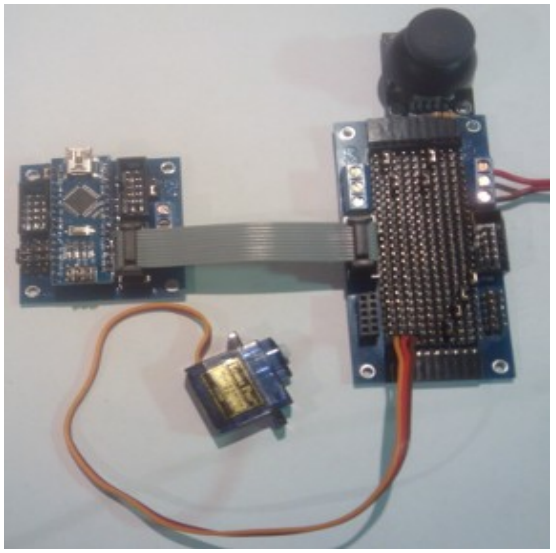
Matriu

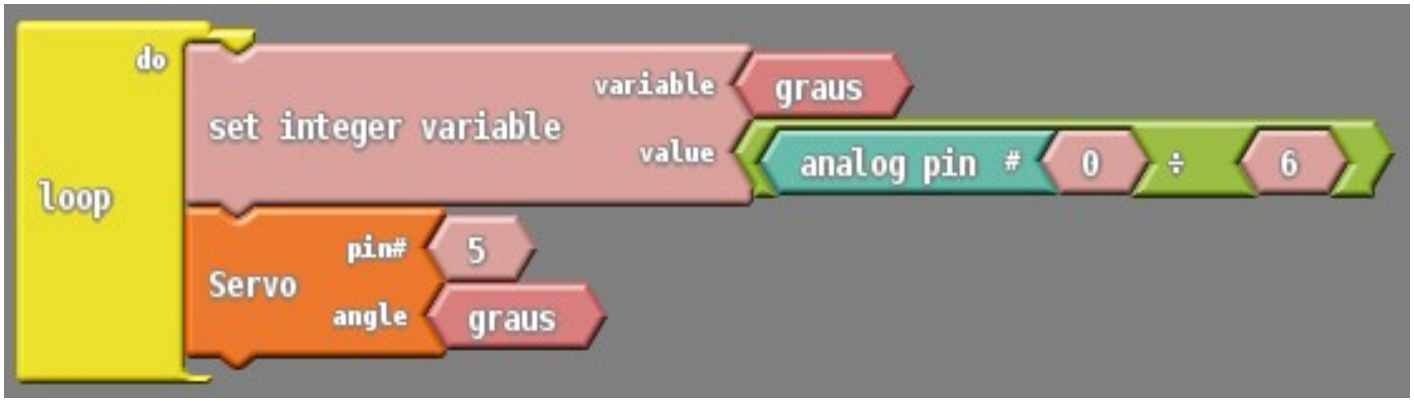


http://electronics.cat/php/common/index.php?lang=ca&page=517

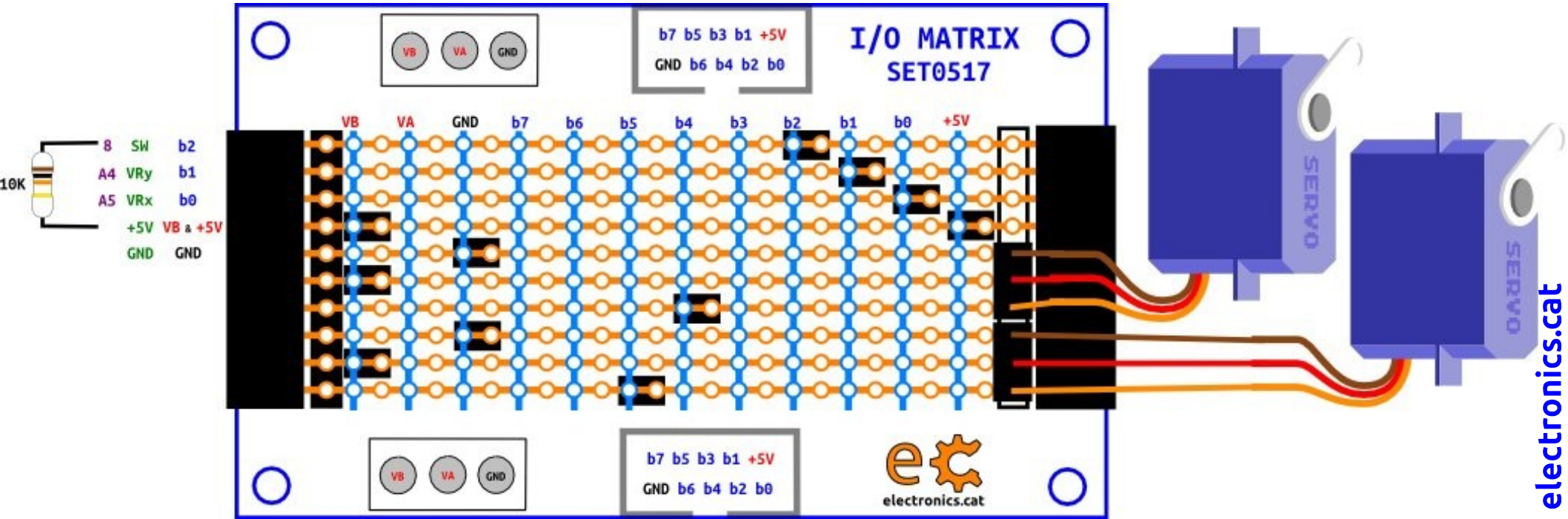
Arduino i Raspberry Pi

Exemple d'ús de la matriu





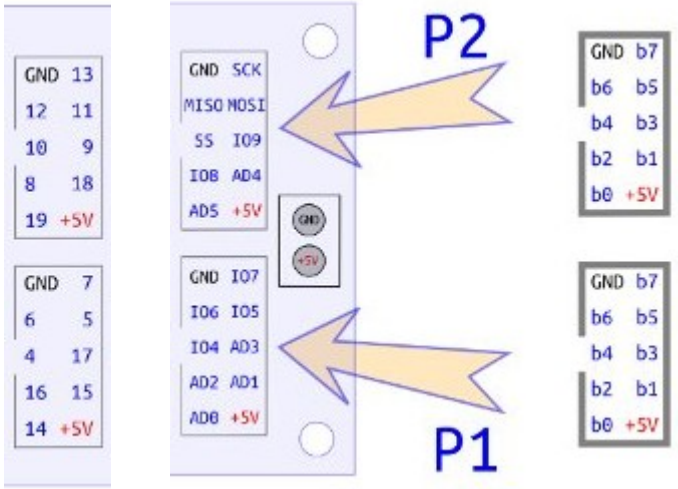
http://www.binefa.cat/php/arduino/ardublock/pr04joystickServo.abp



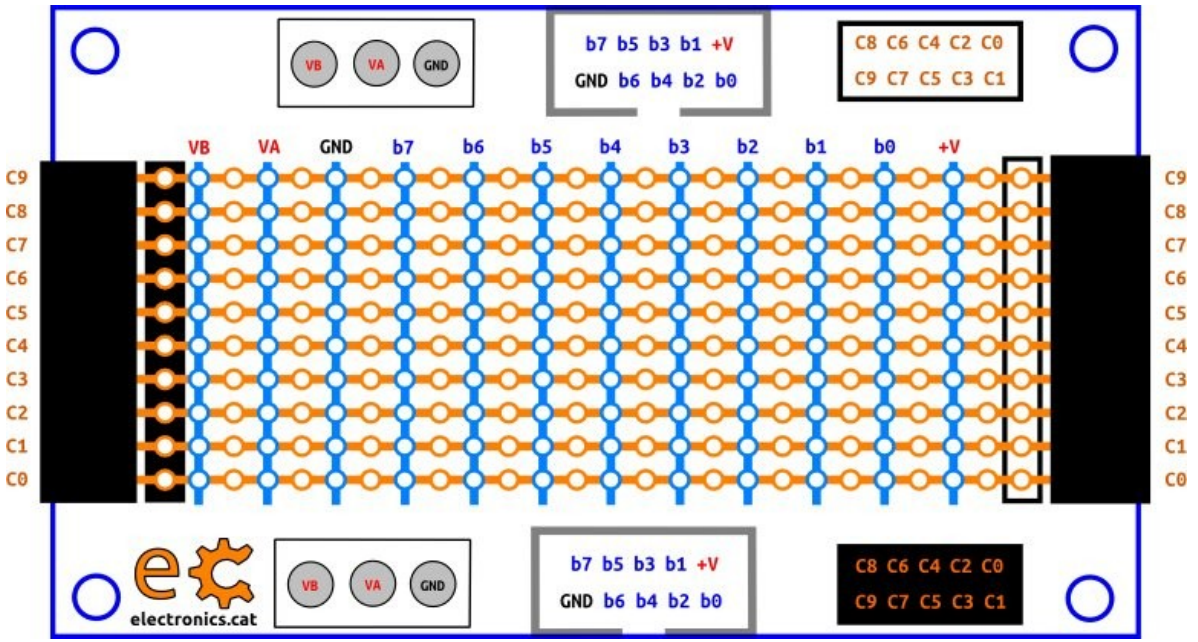
Arduino i Raspberry Pi

Matriu

Connexió a la matriu inversa (pàgina 100 del pdf)				
C8	C6	c4	C2	C0
C9	C7	C5	C3	C1
Connexió de l'ENC28J60				
CLK	WOL	SI	CS	VCC
INT	SO	SCK	RST	GND
Connexió bus ecat				
N	NC	b5	b4	3V3
NC	b6	b7	+5V	GND



Per pota de connector ecat		
1	CK	NC
2	INT	NC
3	WOL	NC
4	SO	b6
5	SI	b5
6	SCK	b7
7	CS	b4
8	RST	+5V
9	VCC	3V3
10	GND	



http://electronics.cat/php/common/index.php?lang=ca&page=517

Arduino i Raspberry Pi

Ethernet – Recepció de datagrames UDP

```
#include <UIPEthernet.h>
#include <ecat.h>

EthernetUDP udp;
Ecat ecat;

void setup() {

  Serial.begin(9600);

  uint8_t mac[6] = {0x00,0x01,0x02,0x03,0x04,0x05};

  Ethernet.begin(mac,IPAddress(192,168,1,108));
  //Ethernet.begin(mac,IPAddress(172,20,1,168));

  int success = udp.begin(5000);

  Serial.print("initialize: ");
  Serial.println(success ? "success" : "failed");
  ecat.setupHighNibbleP1(OUTPUT);

}
```

```
void loop() {

  //check for new udp-packet:
  int size = udp.parsePacket();
  if (size > 0) {
    do
      {
        char* msg = (char*)malloc(size+1);
        int len = udp.read(msg,size+1);
        msg[len]=0;
        Serial.print("received: '");
        Serial.print(msg);
        if(msg[0]=='0'){
          digitalWrite(ecat.nPinP1B7, LOW);
        }else{
          if(msg[0]=='1'){
            digitalWrite(ecat.nPinP1B7, HIGH);
          }
        }
        free(msg);
      }
    while ((size = udp.available())>0);
    //finish reading this packet:
    udp.flush();
    Serial.println("'");
```

Codi per a Arduino UdpServer01b

Arduino i Raspberry Pi

Ethernet – Enviament de datagrames UDP

```
    int success;
    do
      {
        Serial.print("remote ip: ");
        Serial.println(udp.remoteIP());
        Serial.print("remote port: ");
        Serial.println(udp.remotePort());
        //send new packet back to ip/port of client. This also
        //configures the current connection to ignore packets from
        //other clients!
        success = udp.beginPacket(udp.remoteIP(),udp.remotePort());
        Serial.print("beginPacket: ");
        Serial.println(success ? "success" : "failed");
    //beginPacket fails if remote ethaddr is unknown. In this case an
    //arp-request is send out first and beginPacket succeeds as soon
    //the arp-response is received.
      }
    while (!success);

    success = udp.println("hello world from arduino");

    Serial.print("bytes written: ");
    Serial.println(success);

    success = udp.endPacket();

    Serial.print("endPacket: ");
    Serial.println(success ? "success" : "failed");

    udp.stop();
    //restart with new connection to receive packets from other clients
    Serial.print("restart connection: ");
    Serial.println (udp.begin(5000) ? "success" : "failed");
  }
}
```

Codi per a Arduino UdpServer01b

Arduino i Raspberry Pi

Ethernet – Enviament de datagrames UDP

```
$ ./udp0n.py 192.168.1.108
```

```python
#!/usr/bin/python

import socket
import sys

szServer = str(sys.argv[1])
nUdpPort = 5000

client_socket = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
client_socket.sendto("1", (szServer,nUdpPort))
client_socket.close()
```

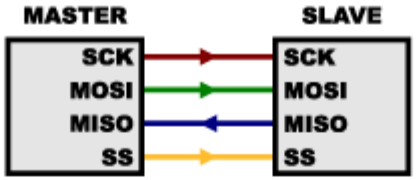
```
$ ./udp0ff.py 192.168.1.108
```

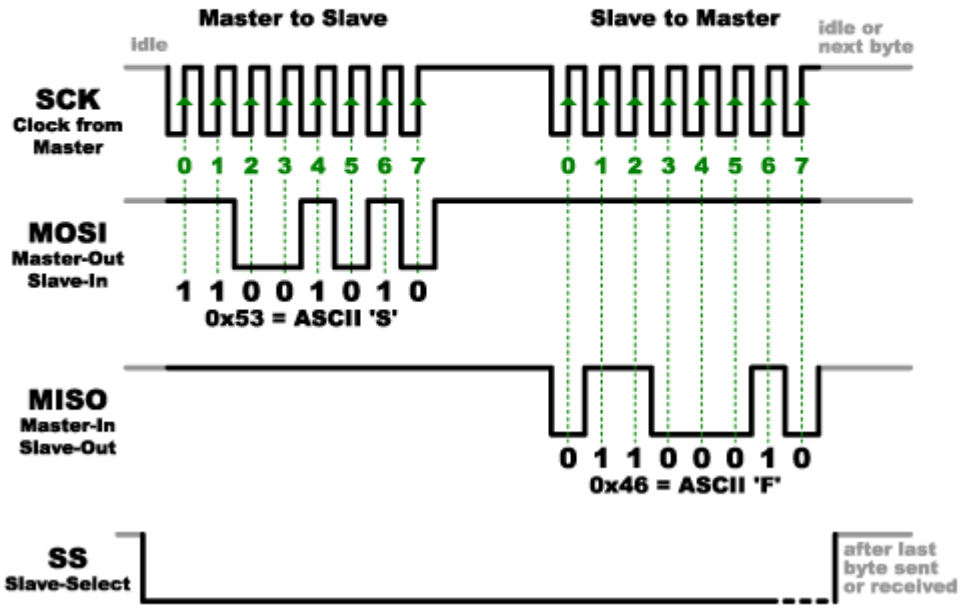
```python
#!/usr/bin/python

import socket
import sys

szServer = str(sys.argv[1])
nUdpPort = 5000

client_socket = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
client_socket.sendto("0", (szServer,nUdpPort))
client_socket.close()
```

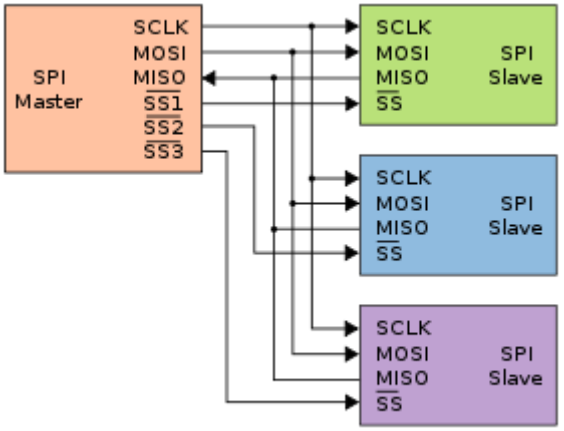
Codis font dels codis d'enviament de datagrames UDP en Python

Arduino i Raspberry Pi

SPI







Explicació del bus SPI

Comunicacions amb Raspberry Pi

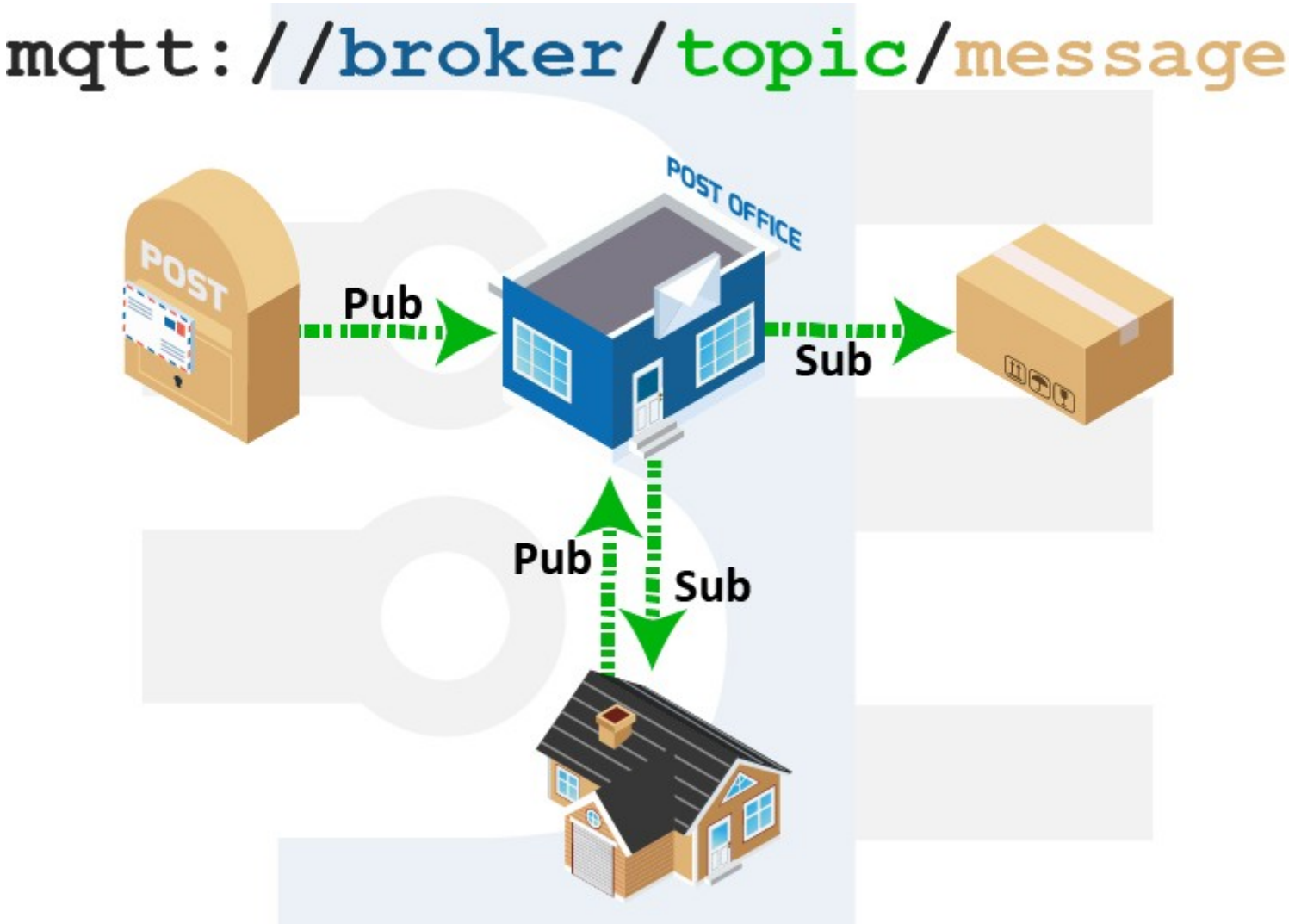
MQTT – Què és?

MQTT és una forma de comunicar dispositius d'Internet de les Coses (IdC / IoT) entre si. És un protocol lleuger i molt versàtil que es pot fer servir des d'un Arduino, una Rasperry Pi, un PC multinucli fins als serveis d'Amazon Web Services.

https://www.baldengineer.com/mqtt-introduction.html
http://binefa.cat/php/dam/m09uf3/20170224/2017024_mqtt_00.pdf

Comunicacions amb Raspberry Pi

Analogia amb el servei postal



https://www.baldengineer.com/mqtt-introduction.html

Comunicacions amb Raspberry Pi

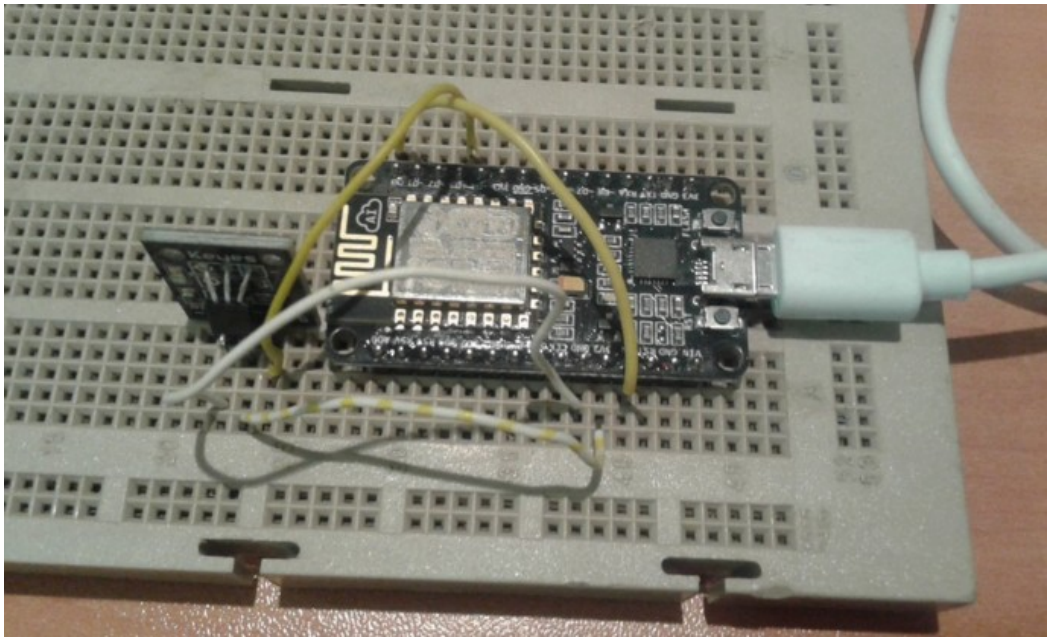
Perquè no fer servir HTTP (o REST)

HTTP requereix de múltiples accions POST per a distribuir un missatge a més d'un client. L'objectiu del sistema de missatgeria intermediària és que l'intermediari distribueixi el missatge, tan sols als clients interessats. La funcionalitat del MQTT és trametre d'un a molts.

https://www.baldengineer.com/mqtt-introduction.html

Comunicacions amb Raspberry Pi

ESP8266 sobre l'IDE d'Arduino - MQTT



```
Fitxer  Edita  Vi
Client mosqsub
Client mosqsub
24.94
Client mosqsub
25.00
Client mosqsub
24.94
Client mosqsub
25.00
Client mosqsub
Client mosqsub
Client mosqsub
25.00
Client mosqsub
25.06
^C
ecat@debian8:~$ mosquitto_sub -d -t sensors/test/temperature
Client mosqsub/15509-debian8 sending CONNECT
Client mosqsub/15509-debian8 received CONNACK
Client mosqsub/15509-debian8 sending SUBSCRIBE (Mid: 1, Topic:
Client mosqsub/15509-debian8 received SUBACK
Subscribed (mid: 1): 0
Client mosqsub/15509-debian8 received PUBLISH (d0, q0, r0, m0,
24.81
Client mosqsub/15509-debian8 received PUBLISH (d0, q0, r0, m0,
24.75
Client mosqsub/15509-debian8 received PUBLISH (d0, q0, r0, m0,
24.69
```

```
Entering deep sleep mode for 3 seconds...
rl  l@@|  @l@|◁   @ l@ b|@@◁@,@r@b@ b@@nn@lnn@@@ b⊙p@@lrlrlp@n
Connecting to IoT-eCat
.............
WiFi connected
IP address:
192.168.1.12
Attempting MQTT connection...connected
Requesting DS18B20 temperature...
Sending temperature: 24.69
Closing MQTT connection...
Closing WiFi connection...
Entering deep sleep mode for 3 seconds...
rl  l@@|  @l@|◁   @ l@ b|@@◁@,@r@b@ b@@nn@lnn@@@ b⊙p@@lrlrlp@n
Connecting to IoT-eCat
....
☑ Desplaçament automàtic
```

http://www.jerome-bernard.com/blog/2015/10/04/wifi-temperature-sensor-with-nodemcu-esp8266/

http://binefa.cat/php/dam/m09uf3/20170224/esp8266_03.pdf

Torn de preguntes ...

Arduino i Raspberry Pi

Presentació descarregable a : **http://binefa.cat/blog**

Correu electrònic de contacte : **jordibinefa@electronics.cat**

twitter

https://twitter.com/JordiBinefa

https://twitter.com/electronicscat

LinkedIn **http://es.linkedin.com/pub/jordi-binefa/13/717/90b**

Plaques disponibles a :

http://www.electronics.cat

http://www.makeit.cat

onda radio, s.a.

NUFESA ELECTRONICS

Moltes gràcies per la vostra atenció