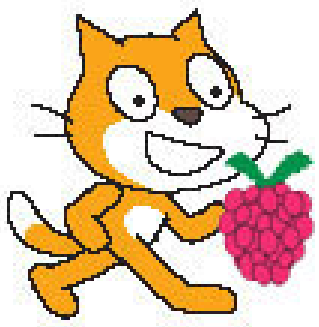
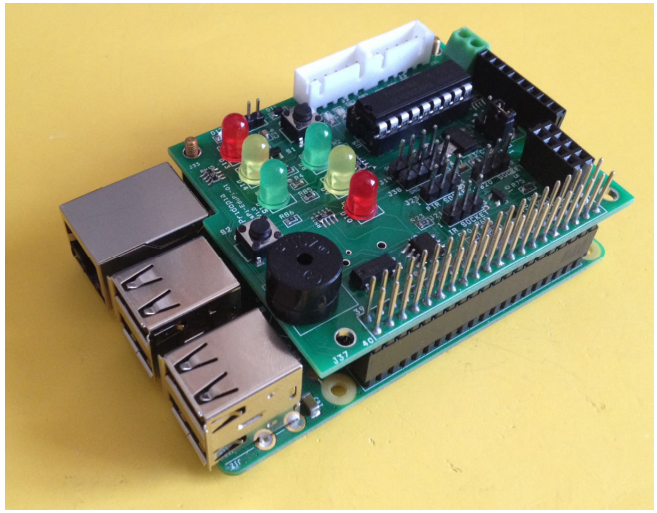
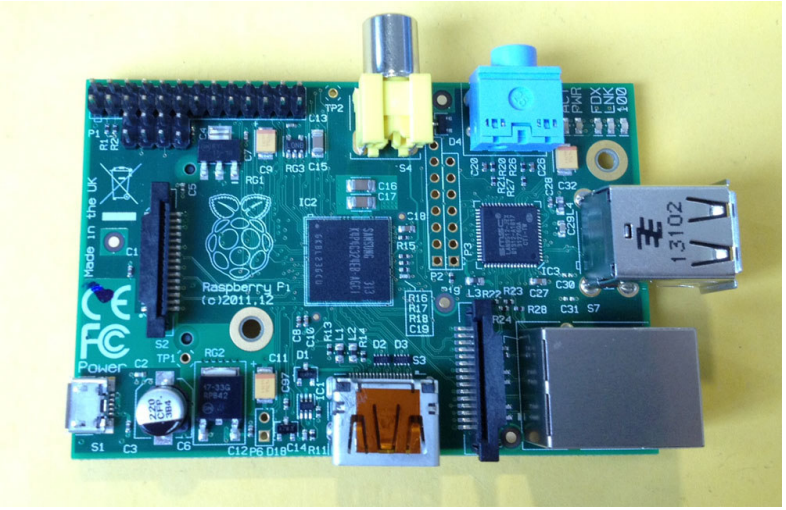
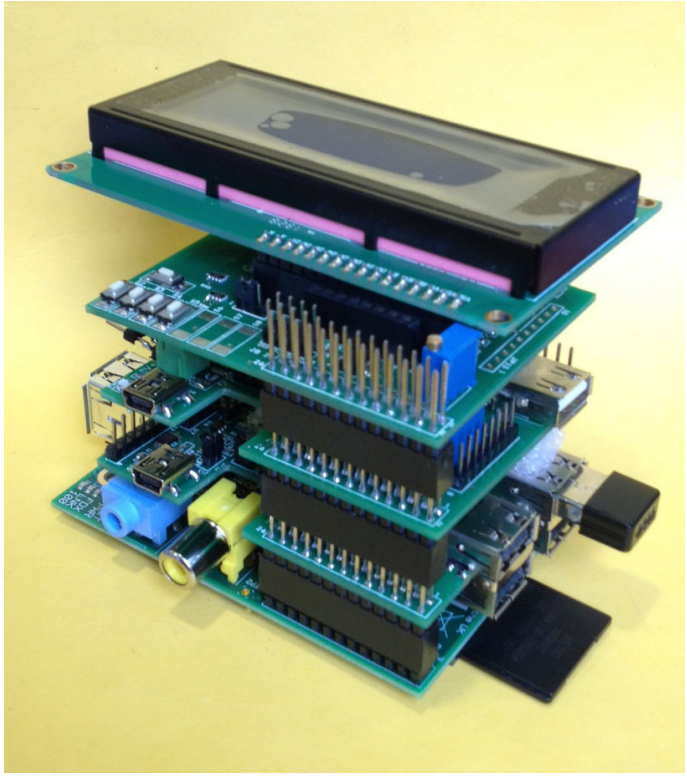




Scratch Raspberry Pi control

Pi Scratch V 2.73 B+ User Manual





Pridopia Limited

Tony Chang

29,Sep,2014

Contents

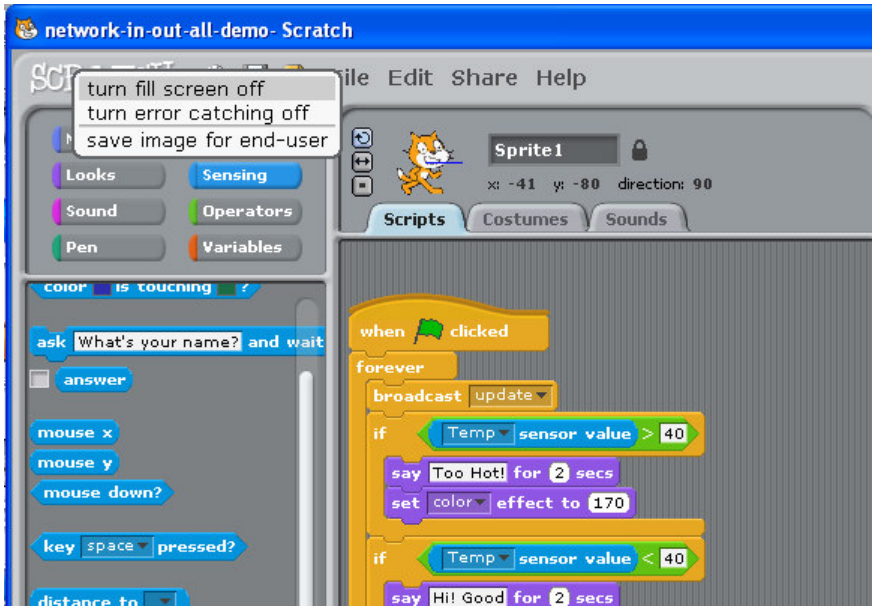
1. Setting Scratch to active Mesh function ---------- 1
2. Download & Install Pi_Scratch software ----------7
Setting Scratch for Auto loading Pi_Scratch ---------- 9
3. Enable i2c & Install i2c ---------- 10
4. Other software pack install ---------- 12
5. Basic GPIO Function ---------- 18
1 BASIC GPIO input & output demo ---------- 18
2 BASIC GPIO step motor demo ---------- 20
3 BASIC GPIO DC motor demo ---------- 21
4 BASIC GPIO Servo motor demo ---------- 22
6. Demo 1 LED control & 2 DC Motor control ---------- 24
(1) LED output Test include (GPIO 28,29,30,31) ---------- 24
(2) 2 Motor forward, Backward, Left, Right control test ---------- 25
1.Motor On Off + Speed control ---------- 26
2. BASIC GPIO Motor on/off control ---------- 27
7. Demo 2 5 GPIO Switch & 16 PWM & 4 Motor Board demo ---------- 28
(1) 5 GPIO Switch input GPIO 7,8,9,10,11 demo ---------- 28
(2) PWM / Servo control demo ---------- 29
8. Demo 3 4hub/7hub i2c function board demo ---------- 31
(1) i2c address 48 TMP102 temperature sensor test ---------- 32
(2) i2c address 68 DS1307 RTC detect & demo ---------- 33
(3) i2c address 20 23008 8 GPIO test ---------- 35
(4) i2c address 49 ADC ads1015 4 channel input test ---------- 36
(5) i2c address 50 EEPROM 24c32 read/write ---------- 37
(6) i2c address 60 DAC MCP4725 demo ---------- 39
9. Demo 4 Stepper Motor and Relay control ---------- 40
10. Demo 5 Input & output test ---------- 42
11. Demo 6 Input & i2c 23017 16 GPIO & 16 LED output test ---------- 44
12. Demo 7 i2c 16 channel PWM & 23017 16 GPIO output test ---------- 47
(1) Servo Motor control ---------- 47
(2) PWM LED control ---------- 48
13. Demo 8 i2c 32 channel PWM /Servo Test ---------- 50
14. Demo 9 i2c 23017-4 64 GPIO & 64 LED output test ---------- 52
15. Demo 10 i2c 23017-2 32 GPIO & 32 LED output test ---------- 54
16. Demo 11 23017-2803 Switch input test ---------- 56
17. Demo12 spi 23S17 32/64 GPIO demo ---------- 58
(1) spi 23s17-4 64 GPIO & 64 LED output test ---------- 59
(2) spi 23s17-2 32 GPIO & 32 LED output test ---------- 59
18. Demo 13 spi 23s17-8 128 GPIO ---------- 60
19. Demo 14 Input &spi 23s17 16 GPIO & 16 LED output test ---------- 62
20. Demo 15 SPI MCP3002 & MCP4802 AD/DA demo ---------- 64
(1) MCP3002 ADC 10 bit Analog-to-Digital Converter ---------- 64
(2) MCP4802 DAC 8 bit Digital-to-Analog Converter ---------- 65
21. Demo 16 i2c 18bit MCP3424 ADC demo ---------- 66
22. Demo 17 i2c LED 24x16 & 16x16 Matrix ---------- 69
(1) i2c 24x16 LED matrix ---------- ---------- 69
(2) i2c 16x16 LED matrix ---------- 71
23. Demo 18 GPIO 84x48 LCD , 16x2 LCD, 20x4 LCD ---------- 72
(1) 5 GPIO Switch GPIO 7,8,9,10,11 ---------- 72
(2) 84x48 pixels monochrome LCD ---------- 73
84x48 LCD RSS function ---------- 74
84x48 LCD new command ---------- 76
(3) 20x4 Character LCD ---------- 77
20x4 LCD RSS function ---------- 78
20x4 LCD new command ---------- 81
(4) 16x2 Character LCD ---------- 81
16x2 LCD RSS function ---------- 82
16x2 LCD new command ---------- 83
24. Demo 19 GPIO 84x48 LCD , i2c 16x2 LCD, i2c 20x4 LCD ---------- 84
(1) i2c 20x4 Character LCD ---------- 84
i2c 20x4 LCD RSS function ---------- 86
(2) i2c 16x2 Character LCD ---------- 87
i2c 16x2 LCD RSS function ---------- 89
(3) i2c LCD1602 & LCD2004 8 switch detect ---------- 90
(4) i2c 20x4 LCD new command ---------- 91
(5) multi i2c 20x4 LCD ---------- 92
25. Demo 20 All Kinds of Sensors ---------- 93
(1) 1-Wire DS18B20 temp Sensor ---------- 93
(2) BMP085 Sensor ---------- 95
(3) Ultrasonic distance sensor (HY-SRF05) ---------- 97
(4) DHT22/AM2302 Temperature & Humidity Sensor ---------- 99
(5) Trace Line Hunting IR Sensor ---------- 101
Connect to I2c GPIO ---------- 104
(6) IR Obstacle Avoidance Sensor module ---------- 106
(7) IR Flame Sensor ---------- 109
(8) IR PIR motion Sensor ---------- 111
Connect to I2c MCP23008 GPIO ---------- 113
(9) IR Remote control ---------- 115
(10) Motor wheel Encoder ---------- 119
(11) I2c MCP23017 GPIO Wheel Encoder ---------- 121
(12) I2c MPU6050 3 Axis Gyroscope + Accelerometer Module ---------- 123
26. Demo 21 i2c MCP23017 Smotor command ---------- 125
27. Demo 22 GPIO LED control ---------- 127
28. Demo 23 Maplin Robotic Arm demo ---------- 128
(1) xWindows control mode ---------- 132
(2)Scratch GPIO control mode ---------- 132
(3)Scratch PWM command mode ---------- 134
29. Demo 24 Servo Robot Arm demo ---------- 137
30. Demo 25 RFID reader & Step Motor & Relay control ---------- 140
31. Raspberry Pi B+ GPIO support ---------- 145
32. Raspberry Pi B+ 8 DC Motor PWM & GPIO support ---------- 149
33. Raspberry Pi B+ EDU Learning Board support ---------- 151

Q & A

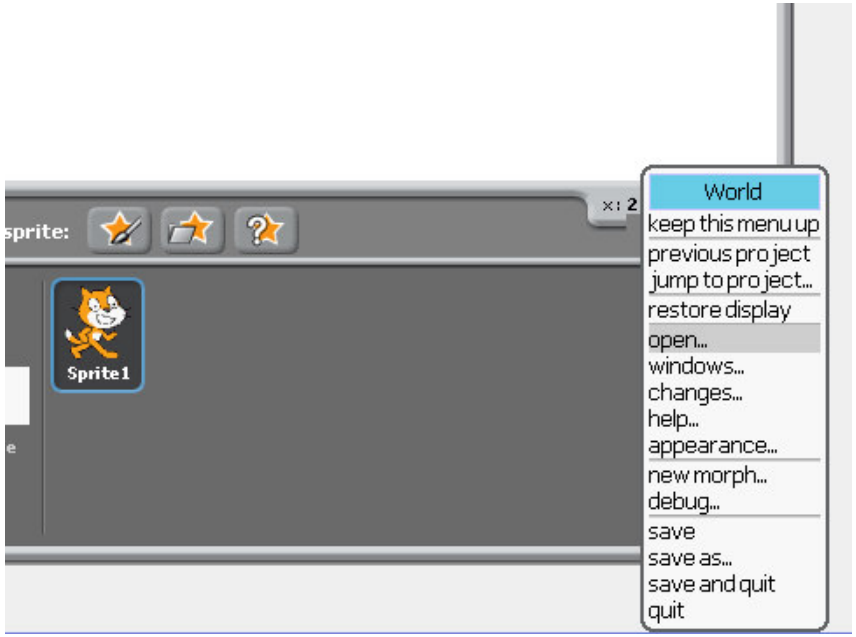
Pi_Scratch Install Q & A ---------- 153
How to Auto run Scratch & auto load Pi_Scratch example file ---------- 157
How to make Image from three ZIP file ---------- 159
First thing need do for Pi_Scratch ready Image ---------- 160
How to upgrade from old Pi_Scratch to new Ver Pi-Scratch ---------- 161
VNC (VIRTUAL NETWORK COMPUTING) setting ---------- 162

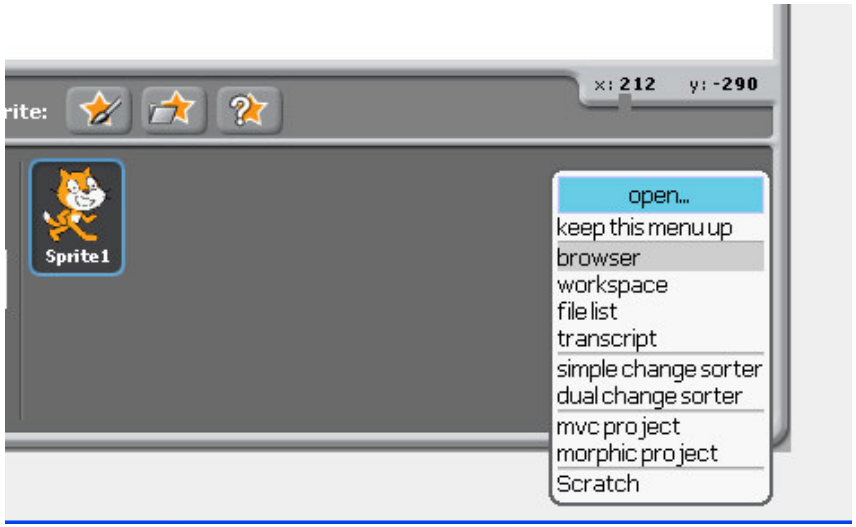
1. Scratch V1.4 Setting to Active Mesh function

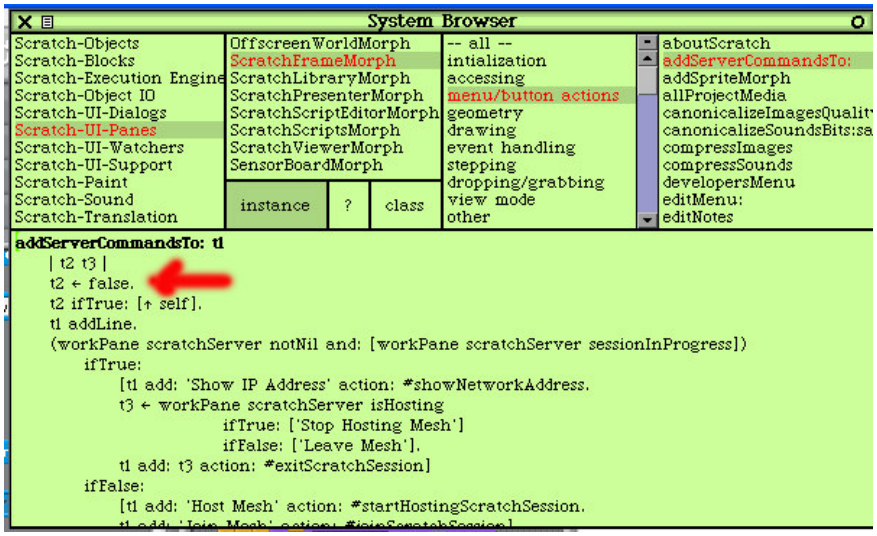
1. Shift-click the upper part of the R in the Scratch logo.



2. Select "turn fill screen off".
3. You will now have a white area at the bottom and right side of the screen. Click on the white area and select "open...".
4. Now select "browser".
5. Now you've opened the System Browser! In this place you can edit Scratch.





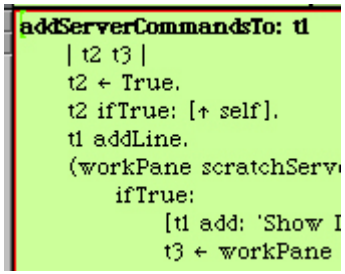


5. Now you've opened the System Browser! In this place you can edit Scratch.

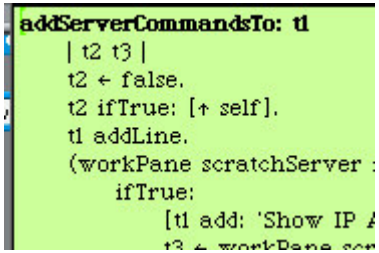
Go to Scratch-UI-Panes - ScratchFrameMorph - menu/button actions - addServerCommandsTo: .

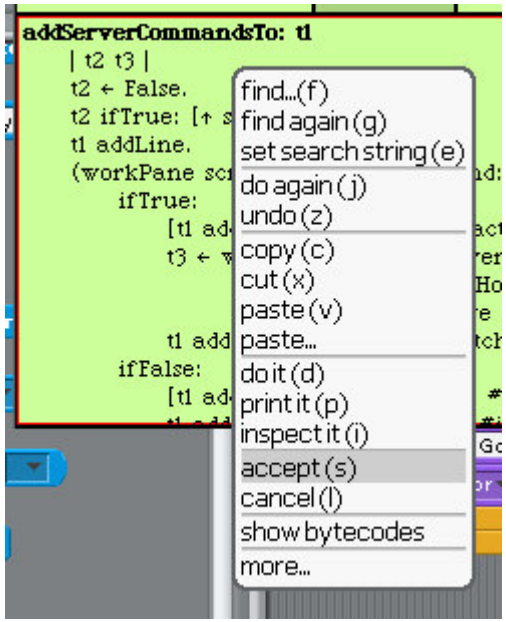
6. Look down, and edit the "t2 ← true" to "t2 ← false".
7. Right-click and select "accept".

(or press middle key or wheel)

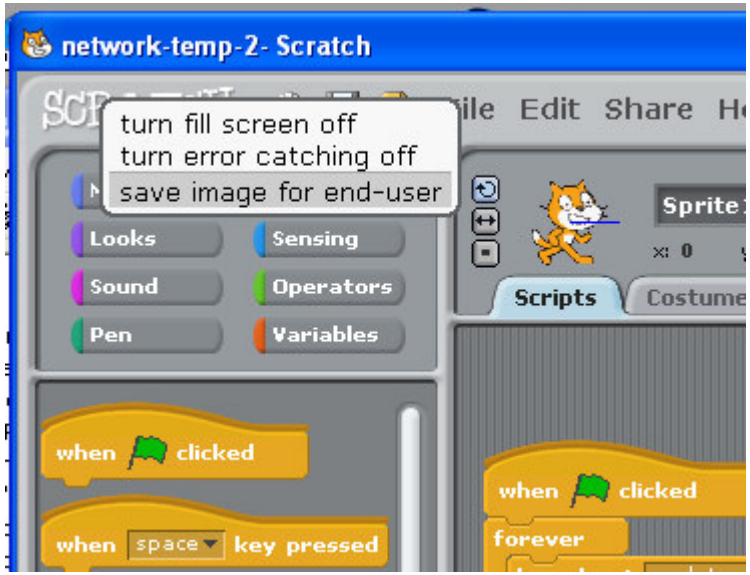


to





8. Enter in your initials (you can fake it, it doesn't matter) and click "accept".
9. Exit the System Browser.
10. Shift-click the R and select "turn fill screen on".



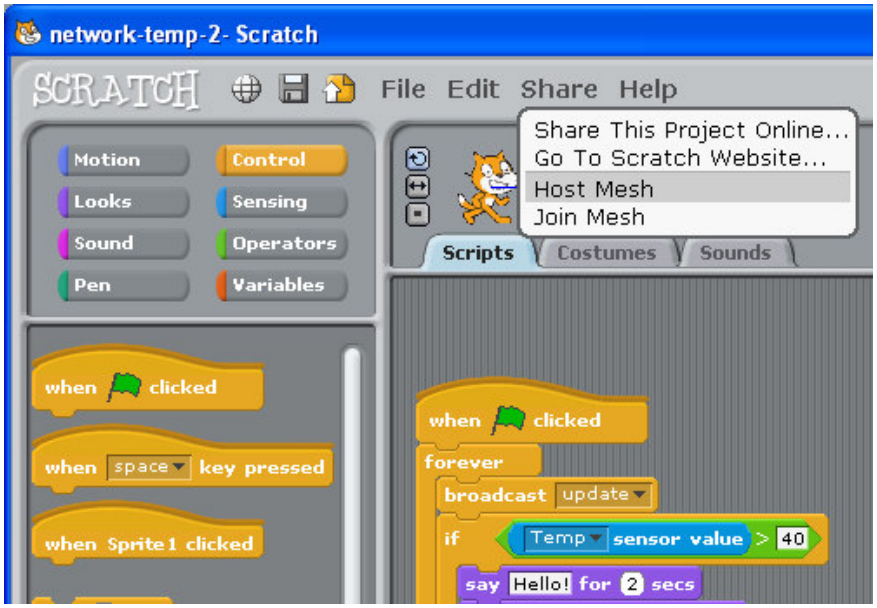
If you don't want to go through the process every time you want to use Mesh, you can save the System Browser so Mesh will stay available - Shift-click the R again and select "save image for end-user".

If you don't want Mesh anymore, just change the 'false' back to 'true'.

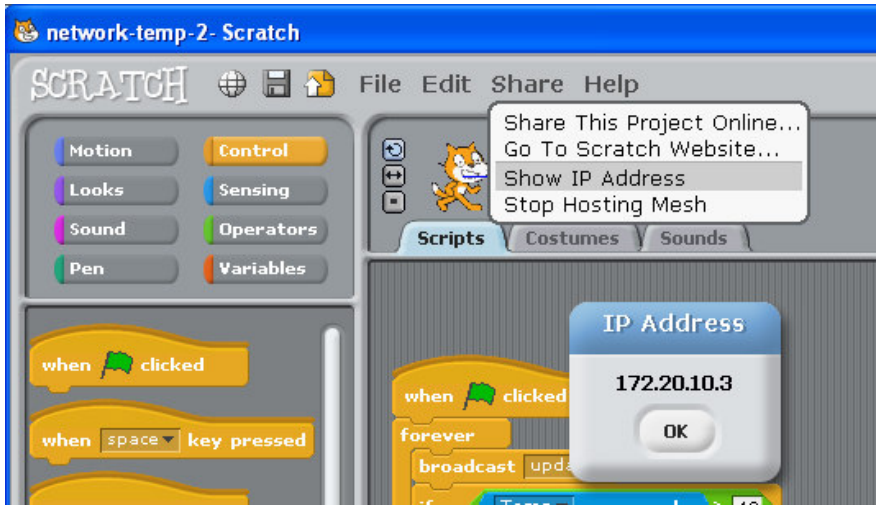
Active Mesh

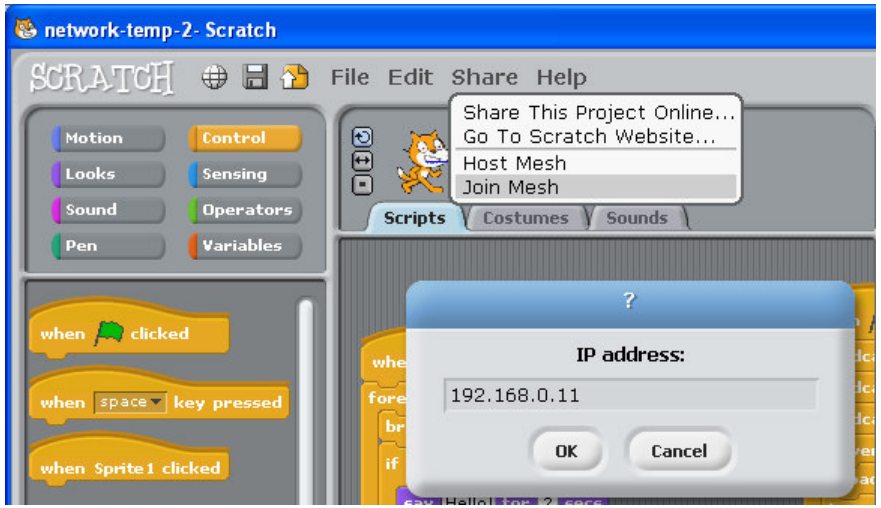
- **Host Mesh lets you host a Mesh session.**
- **Join Mesh lets you join a Mesh session.**

To host a Mesh session:
1. Click Host Mesh.
2. Done! Your IP address will show up



What if you want to join a Mesh session, but the person hosting the Mesh session forgot to tell you the IP address, and he/she forgot what the IP address is? He/she Shift-clicks the Share button and selects "Show IP address". The IP address will pop up like before.





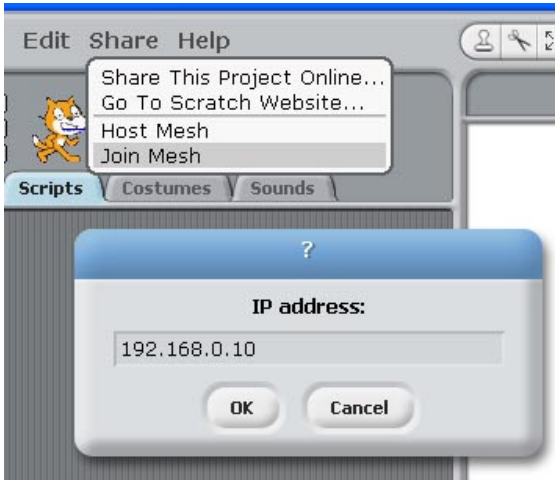
To join a Mesh session:
1. Click Join Mesh.
2. Enter in the IP address of the Raspberry Pi that is hosting the Mesh session. (use "ifconfig" in terminal command line)

A Mesh session can have more than two Scratch programs connected at once - there's no limit! All Scratch programs have to connect to the same IP though.

Use "ifconfig" check the IP of Raspberry PI

```
192.168.0.10 - PuTTY
eth0      Link encap:Ethernet  HWaddr b8:27:eb:3d:c1:b1
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:64 errors:0 dropped:0 overruns:0 frame:0
          TX packets:64 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:4209 (4.1 KiB)  TX bytes:4209 (4.1 KiB)

wlan0     Link encap:Ethernet  HWaddr 00:25:22:4a:fe:41
          inet addr:192.168.0.10  Bcast:192.168.0.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:847 errors:0 dropped:0 overruns:0 frame:0
          TX packets:243 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:172694 (168.6 KiB)  TX bytes:29504 (28.8 KiB)

root@raspberrypi:~#
```

Join Mesh "input Raspberry Pi IP" login from Desktop or Laptop



Download program pack from our web site

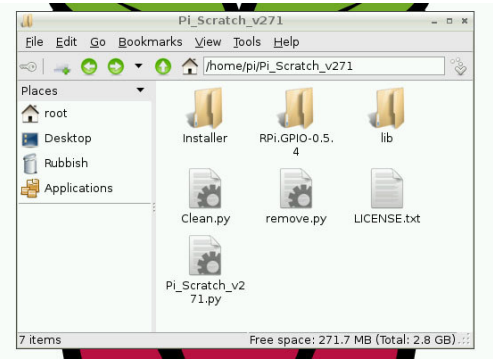
sudo wget http://pridopia.co.uk/pi-pgm/Pi_Scratch_v273.tar

or sudo wget http://pridopia.com/pi-pgm/Pi_Scratch_v273.tar

tar xf Pi_Scratch_v273.tar

cd Pi_Scratch_v273

```
192.168.0.20 - PuTTY
root@raspberrypi:/home/pi# wget http://pridopia.co.uk/pi-pgm/Pi_Scratch_v271.tar
--2014-07-25 10:27:29--  http://pridopia.co.uk/pi-pgm/Pi_Scratch_v271.tar
Resolving pridopia.co.uk (pridopia.co.uk)... 79.170.44.105
Connecting to pridopia.co.uk (pridopia.co.uk)|79.170.44.105|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3369472 (3.2M) [application/x-tar]
Saving to: `Pi_Scratch_v271.tar'

100%[=====================================>] 3,369,472    725K/s   in 7.8s

2014-07-25 10:27:39 (423 KB/s) - `Pi_Scratch_v271.tar' saved [3369472/3369472]

root@raspberrypi:/home/pi# tar xf Pi_Scratch_v271.tar
root@raspberrypi:/home/pi# cd Pi_Scratch_v271
root@raspberrypi:/home/pi/Pi_Scratch_v271# dir
Clean.py   lib          Pi_Scratch_v271.py  RPi.GPIO-0.5.4
Installer  LICENSE.txt  remove.py
root@raspberrypi:/home/pi/Pi_Scratch_v271#
```



You will find inside folder **Pi_Scratch_v273**
have 1 library folder **lib**
one WebIoPi-0.6.0 **support DAC function**
one installer folder include two python tools **Install-With-Scratch.py & Install-Without-Scratch.py** for setting Auto Login & Auto Load Pi_Scratch
Install-autorun-Scratch.py - for auto run example without press any key
one Tools install all necessary program **install.py**
RFID-Installer.py Install RFID tools pack
one License document **LICENSE.txt**
one main program **Pi_Scratch_v273.py**

* set up the auto login & auto start manually.

setting Auto login -> startx -> Scratch -> load file for Pi_Scratch

1. Copy the two [.desktop] files to ~/.config/autostart/ and to the desktop at ~/.Desktop

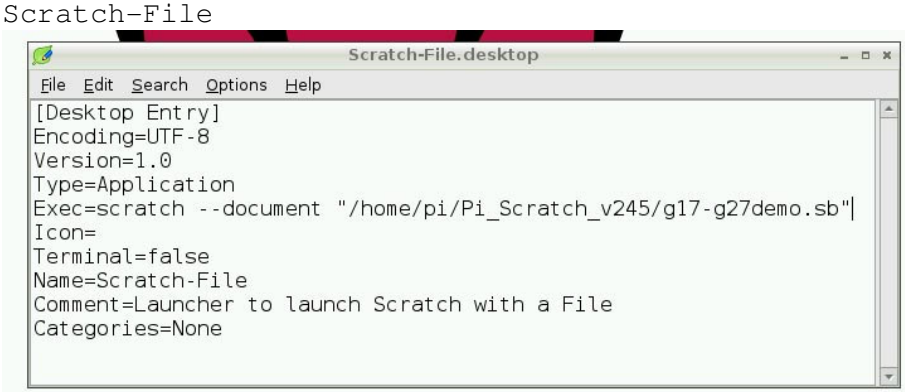
You can use "**sudo python Install-With-Scratch.py** " this will help you create a folder and copy two files to the folder & install Auto login -> startx -> Pi_Sratch

You can use "**sudo python Install-Without-Scratch.py** " this will help you create a folder and copy two files to the folder & install Auto login

Two desktop files you can modify

Scratch-Handler

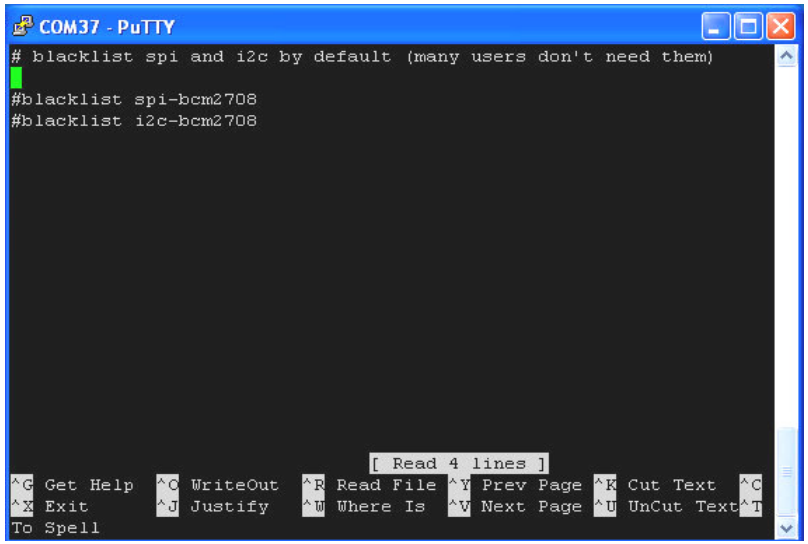
Scratch-File



1.Make sure your I2C driver are enable

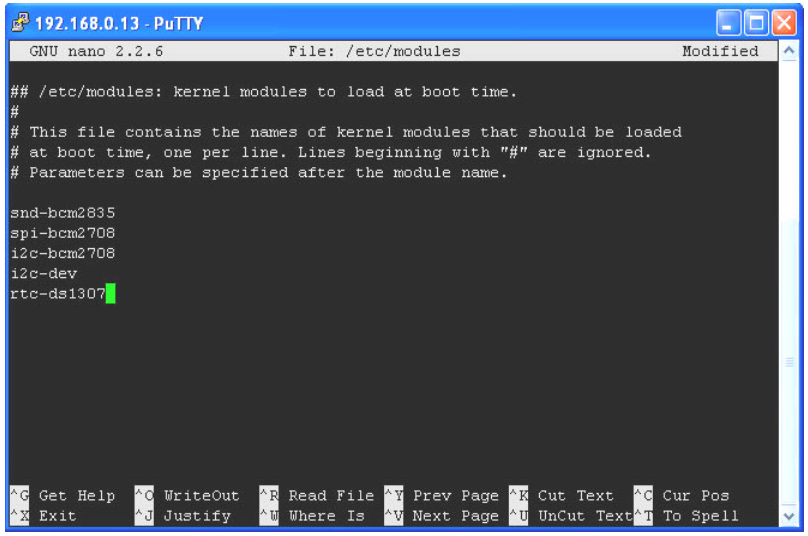
To enable it all you need to do is comment out a line by putting # in front

sudo nano /etc/modprobe.d/raspi-blacklist.conf



2. Add i2c-dev in /etc/modules by use

sudo nano /etc/modules



3. Next install the python-smbus python module:

sudo apt-get install python-smbus

sudo apt-get install i2c-tools

Now you are ready to use the i2c with python.

If you already install I2c driver ,

i2cdetect -y 0 - for Raspberry Pi V1 Board

```
COM12 - PuTTY
raspberrypi login: root
Password:
Last login: Fri Sep 21 17:20:25 GMT 2012 on tty1
Linux raspberrypi 3.1.9adafruit+ #10 PREEMPT Thu Aug 30 20:07:

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

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the ext
permitted by applicable law.

Type 'startx' to launch a graphical session

i2root@raspberrypi:~# i2cdetect -y 0
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:          -- -- -- -- -- -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: 20 -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- 48 49 -- -- -- -- -- --
50: 50 -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: 60 -- -- -- -- -- -- -- 68 -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
root@raspberrypi:~#
root@raspberrypi:~# ~
```

i2cdetect -y 1 - for Raspberry Pi V2 Board

```
COM21 - PuTTY
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the exte
permitted by applicable law.

Type 'startx' to launch a graphical session

i2vroot@raspberrypi:~# i2cdetect -y 1
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:          -- -- -- -- -- -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- 21 22 -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
root@raspberrypi:~#
```

install other necessary software package

use our **install.py** will help you install WebIOPi , spidev & RPIO

sudo python install.py

(1) install WebIOPi

goto folder "WebIOPi-0.6.0" → "Python" →

run the command:

sudo python setup.py install

(2) install spidev

sudo apt-get install git

Then install py-spidev:

git clone git://github.com/doceme/py-spidev

cd py-spidev/

sudo python setup.py install

(3) install GPIO Library

https://pypi.python.org/pypi/RPi.GPIO GPIO library

GPIO library - RPi.GPIO-0.5.6.tar.gz

Install python , library and run the test program

 # sudo apt-get install python-dev

 # wget http://www.pridopia.co.uk/pi-pgm/RPi.GPIO-0.5.6.tar.gz

 # gunzip RPi.GPIO-0.5.6.tar.gz

 # tar –xvf RPi.GPIO-0.5.6.tar

 # cd RPi.GPIO-0.5.6

 # sudo python setup.py install

(4) install **RPIO** 0.10.0

https://pypi.python.org/pypi/RPIO#downloads

 # sudo apt-get install python-setuptools

 # sudo easy_install –U RPIO

Install IR diver by manually

(5) LIRC module

 # sudo apt-get install lirc

(6) python LIRC module

https://pypi.python.org/pypi/python-lirc/1.1

 # sudo apt-get install python-lirc

finished all necessary software install.

sudo Pi_Scratch_v273.py

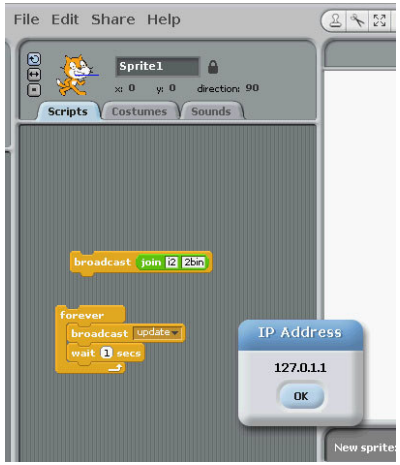
sudo Pi_Scratch_v273.py -d in debug mode with more information display

```
LXTerminal
File Edit Tabs Help
Reseting PCA9685
Setting PWM frequency to 60 Hz
Estimated pre-scale: 100
Final pre-scale: 101
Reseting PCA9685
Setting PWM frequency to 60 Hz
Estimated pre-scale: 100
Final pre-scale: 101
12510
Scratch not up. Sleeping for 5 and trying again.
0x20 not found.
LED x 32 Board loaded MCP23017.
Pridopia LED x 32 Board not loaded MCP23S17.
Reseting PCA9685
Setting PWM frequency to 60 Hz
Estimated pre-scale: 100
Final pre-scale: 101
Reseting PCA9685
Setting PWM frequency to 60 Hz
Estimated pre-scale: 100
Final pre-scale: 101
12510
Scratch not up. Sleeping for 5 and trying again.
```

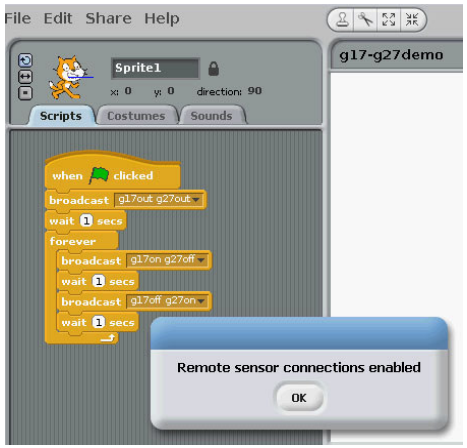
The Program will run and waiting for Scratch connect …

1. Shift-click the Share in menu bar choose "Host Mesh".

2. choose Host Mesh --- will appear a IP address windows



The Program run in back ground detect all the device connect to Raspberry Pi. You can use Scratch with GPIO control function now.



***** For the first time use Pi_Scratch**

Load "g17-g27demo.sb" demo program first, shift-click "Share " → Host Mesh --→ will appear a IP address windows "Ok"

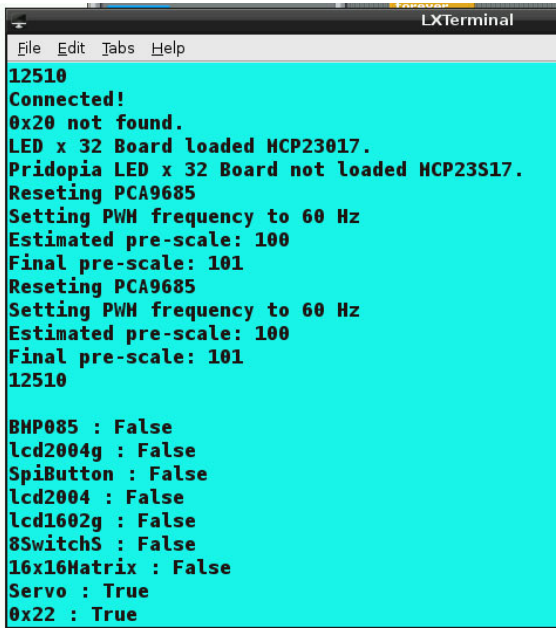
Then save this project in your pi again.

Next time you can load "g17-g27demo" the scratch will automatic connect our "Pi_Scratch" , with "**Remote sensor connections enable**" window, you don't need choose "Host Mesh" every time"

***** You can run Scratch in PC use "Join Mesh"**

To control Raspberry Pi (need IP for your Raspberry Pi)

Raspberry Pi in " Host Mesh"



Minimum Requirements – a Raspberry Pi with Raspbian (wheezy) installed

Our "**Pi_Scratch**" for scratch V1.4 driver

The main goal for this software is towards helping children / beginner programmers / hardware enthusiasts build up their skills by doing simple tasks with even easier commands / controls.

Our Program need you active your scratch Mesh function & support the following products

4hub/7hub i2c RTC & Temperature sensor, AD/DA, EEPROM
i2c 8,16,32,64,128 GPIO board
spi 16,32,64,128 GPIO board
GPIO relay, Motor, stepper Motor , 16x2 ,20x4 , 84x48 LCD Board.
I2c servo board, 16x16 ,24x16 LED matrix
1-Wire DS18B20 Temp Sensor
BMP085 Barometric Pressure/Temperature/Altitude Sensor
Ultrasonic Distance Sensor
I2c MCP3424 ADC , spi mcp3002/mcp4802 AD/DA
DC motor ON/OFF clockwise/anticlockwise & speed
Maplin Motor Robotic ARM
New i2c & GPIO LCD command **** 8,Jan 2014**
DHT22 Digital Temperature & Humidity Sensor **** 14,Jan 2014**
IR- Line Hunting sensor, IR -Flame sensor **** 14,Jan 2014**
6 DOF (Degrees of Freedom) Servo Robot Arm **** 17,Jan 201**
IR- PIR Motion sensor **** 20,Jan 2014**
i2c PWM LED control & GPIO LED control **** 24,Jan 2014**
IR Remote control Set **** 26,Feb 2014**
Motor wheel Encoder **** 05,Mar 2014**
I2c GPIO Motor wheel Encoder **** 14,Mar 2014**
24x16 & 16x16 LED matrix new command update **** 17,Mar 2014**
RFID Reader & Relay Step Motor control **** 30,May 2014**
MPU6050 sensor 3 Axis Gyroscope **** 23,Jul 2014**
8 DC Motor PWM control & RFID reader new update **** 26,Sep 2014**

1.The Pi_Scratch use GPIO number (BCM) not raspberry pi pin number. so can support V2 P5 (GPIO 28,29,30,31) four extra GPIO pins.

2. example 1)define GPIO as input or output

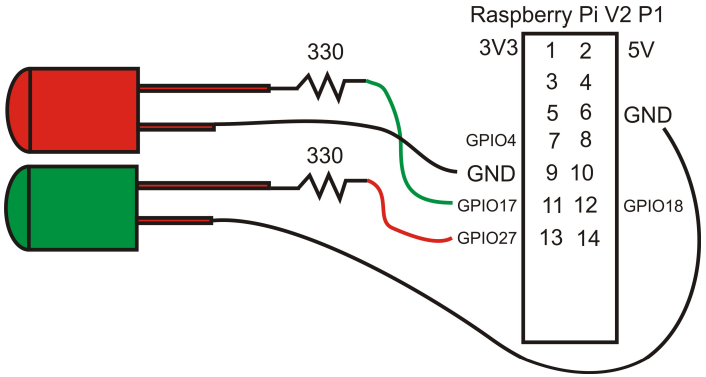
g17out ---> GPIO17 output (P1 pin 11)

g17in --> GPIO17 input

2) set GPIO

g17on --> GPIO17 on(High)

g17off ---> GPIO17 off(Low)



Example : 1.G17out g27out
(g17,g27 LED on off)

Delay 1 sec

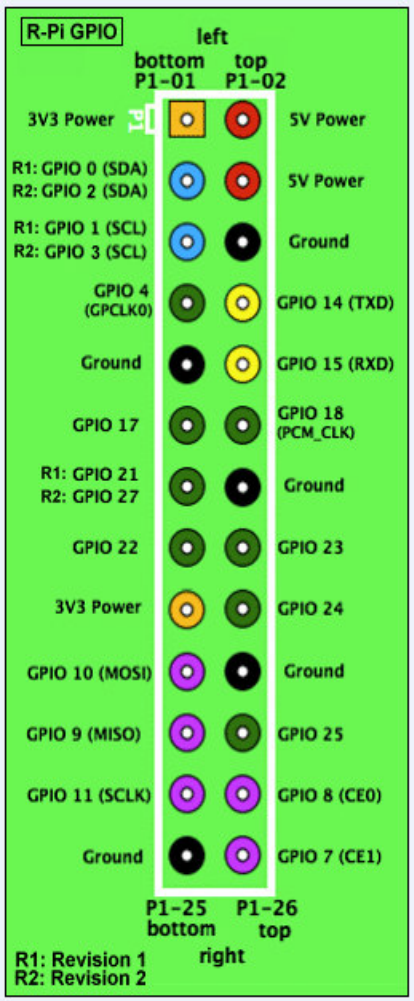
2 G17on g27off

delay 2 sec

3 G17off g27on

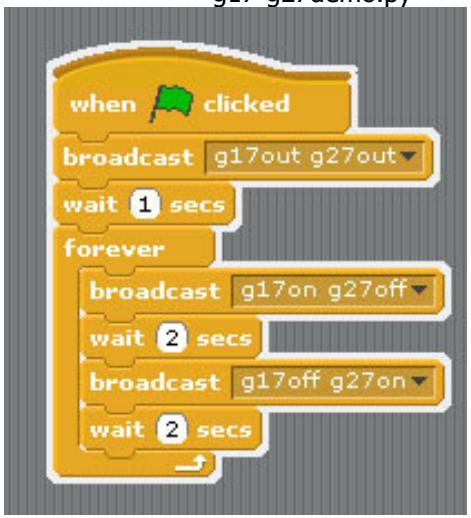
delay 2se

4 goto step 2

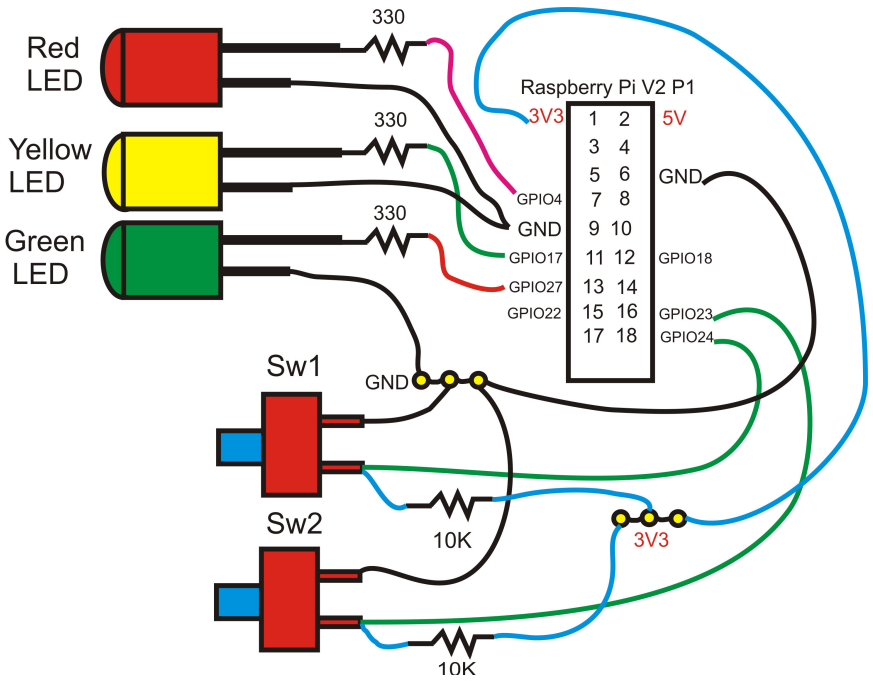


example Program g17-g27demo.sb

g17-g27demo.py

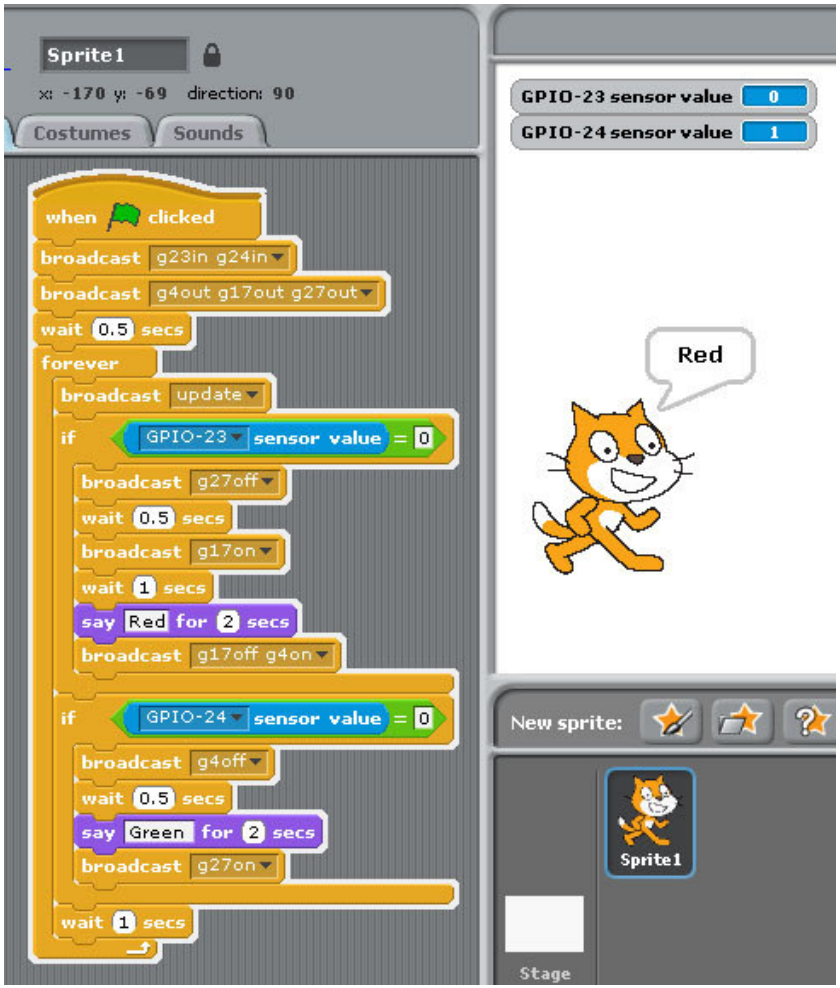


1. BASIC function 1 BASIC GPIO input & output demo

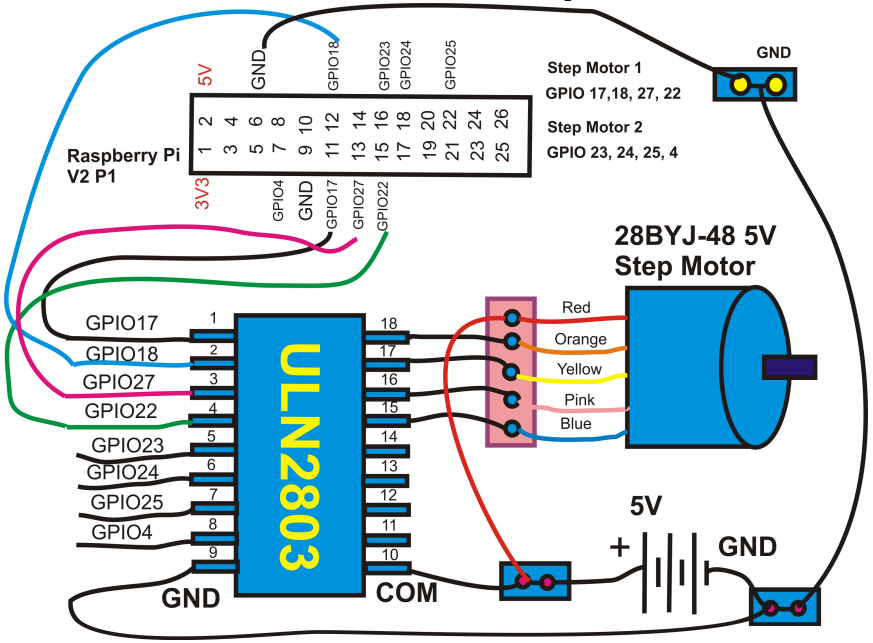


example Program RYG-LED-SW-demo.sb

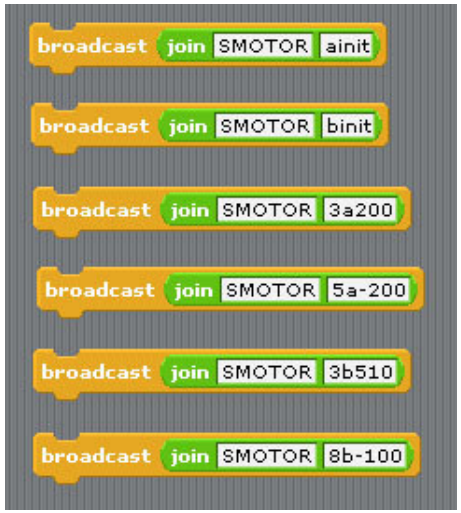
1. define GPIO 23, 24 as input
2.define GPIO 4, 17,27 as output
3. press sw2 yellow LED on 1 sec, then Red LED "ON" The Cat say "Red"
4. press sw1 Red LED off 1 sec, then Green LED "ON" The Cat say "Green"



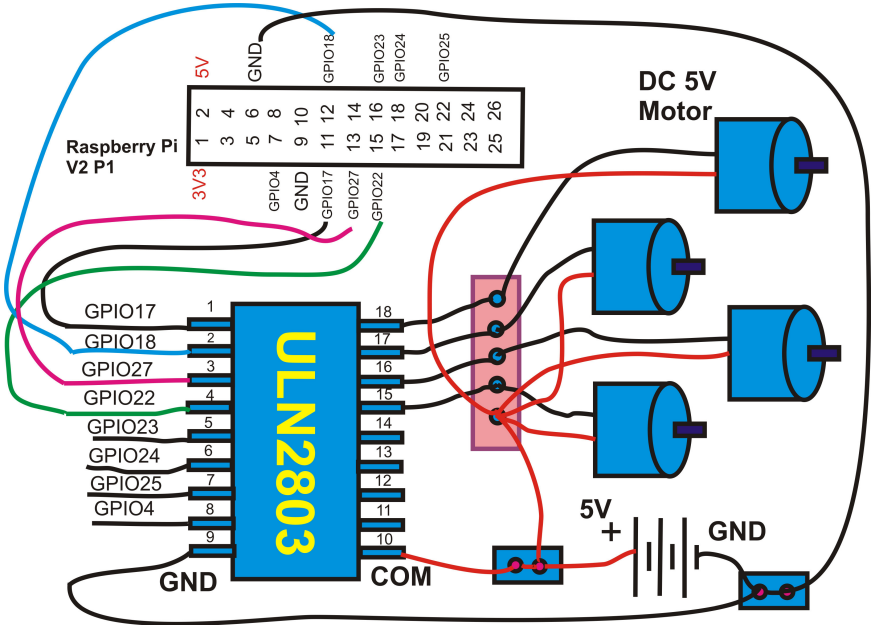
2 BASIC function 2 BASIC GPIO Step Motor demo



command "SMOTORainit" initial GPIO 17,18,27,22 as A Step Motor Port
command "SMOTORbinit" initial GPIO 23,24,25,7 as B Step Motor Port
command "SMOTOR" + speed (3-50) +(a or B) + output step
100 --- clockwise 100 step , -100 --- anticlockwise 100 step
a -- GPIO 17,18,27,22 , b -- GPIO 23,24,25,4



3 BASIC function 3 BASIC GPIO DC Motor demo

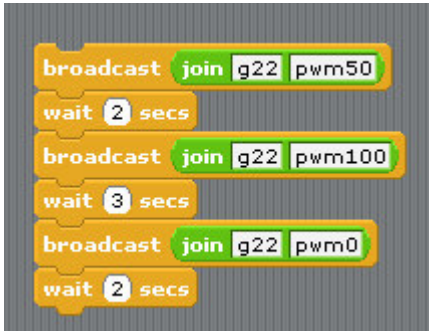


command "GPIO number"+"pwm"+"speed" speed (0 ~ 100)

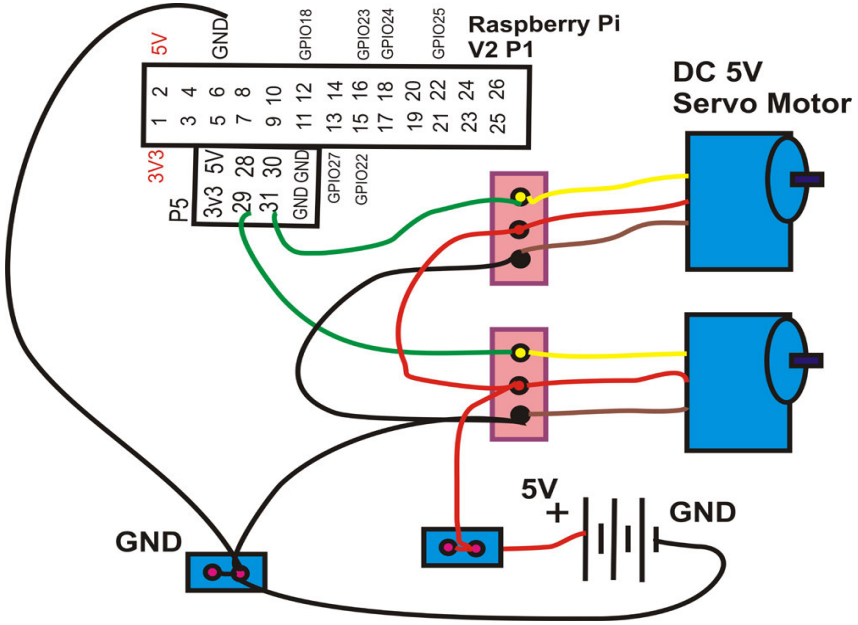
g22pwm50 ==> GPIO 22 ON speed 50

g22pwm100 ==> GPIO 22 ON full speed 100

g22pwm0 ==> GPIO 22 ON speed 0 (STOP)



4 BASIC function 4 GPIO DC Servo Motor control demo



command "**GSE**"+"GPIO number"+"**A**"+"Angle" GSE(**G**PIO **Se**rvo)

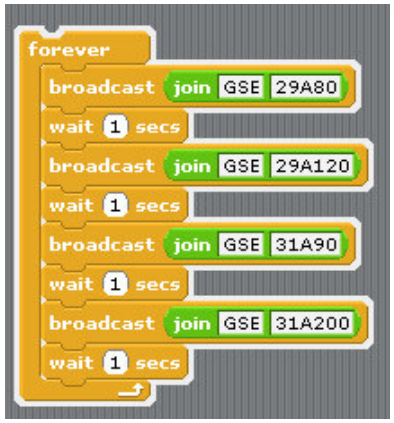
angle degree (50 ~ 350) depend on Servo Motor

GSE29A80 ==> GPIO 29 Servo to 80 degree

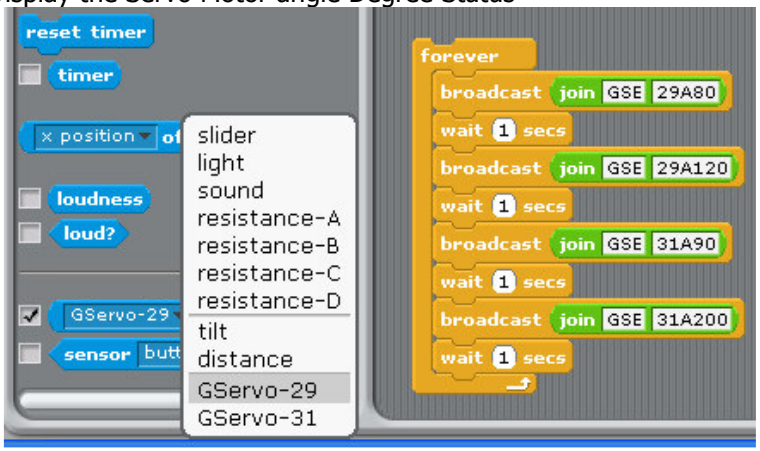
GSE29A120 ==> GPIO 29 Servo to 120 degree

GSE31A90 ==> GPIO 31 Servo to 90 degree

GSE31A200 ==> GPIO 31 Servo to 200 degree



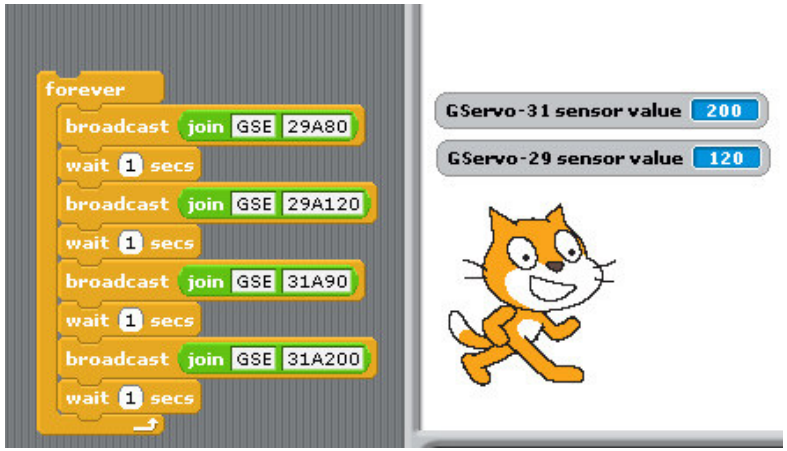
Display the Servo Motor angle Degree Status



after command " GSE29A80 " ==> GPIO 29 Servo to 80 degree

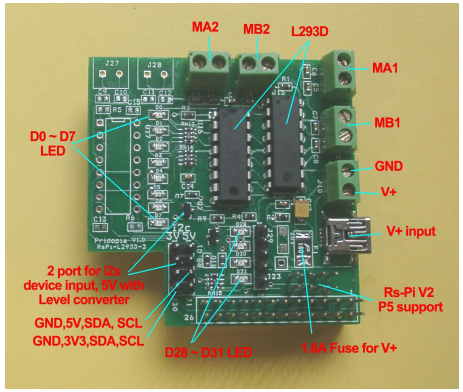
command " GSE31A200 " ==> GPIO 31 Servo to 200 degree

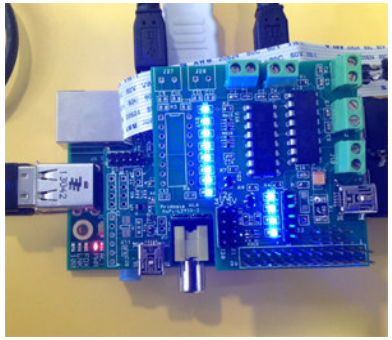
in Sensing --> Slider , you will see the " GServo-29, & GServo-31" in the list



6. Demo1 - LED control & 2 DC Motor control

8GPIO in P1, 4GPIO in P5 output test





GPIO 17,18,27,22,23,24,25,4, --- p1 28,29,30,31 -- v2

p5 output demo use our L293D-2-P5 board, with 8 LED for P1, 4 LED for P5(28,29,30,31)

(1) LED output Test include (GPIO 28,29,30,31)

define GPIO 17,18,27,22,23,24,25,4 & 28,29,30,31 as output

1.send GPIO 17,18,27,22 LED "ON" & 28,29,30, 31 LED "ON"

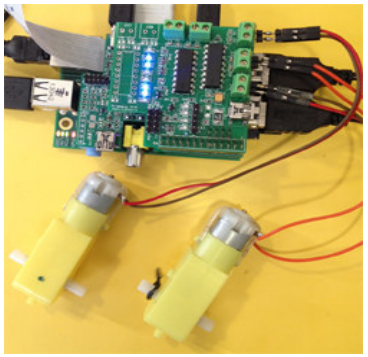
GPIO 23,24,25,4 LED "OFF"

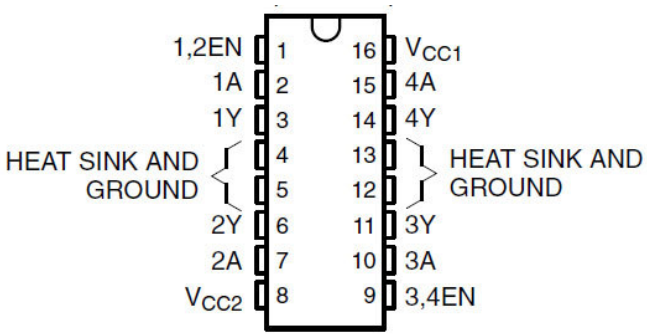
2.send GPIO 17,18,27,22 LED "OFF" & 28,29,30, 31 LED "OFF"

GPIO 23,24,25,4 LED "ON"

3. RETURN to step1

(2) 2 Motor forward, Backward, Left, Right control test





L293D

Pin 1	Pin 2	Pin 7	Function
High	Low	High	Turn clockwise
High	High	Low	Turn anti-clockwise
High	Low	Low	Stop
High	High	High	Stop
Low	Not applicable	Not applicable	Stop

(1) Motor On Off + Speed control

A - GPIO 17,18 Motor A B - GPIO 27,22 Motor B

C - GPIO 23,24 Motor C D - GPIO 25,4 Motor D

command "Motor Name"+ "DM"+"speed"

speed (10 ~100) clockwise

speed (-10 ~ -100) anticlockwise

reset for DMotor (**RDM** – Reset DMotor)

ADM100 DC Motor A full speed 100

BDM100 DC Motor B full speed 100

ADM50 DC Motor A speed 50

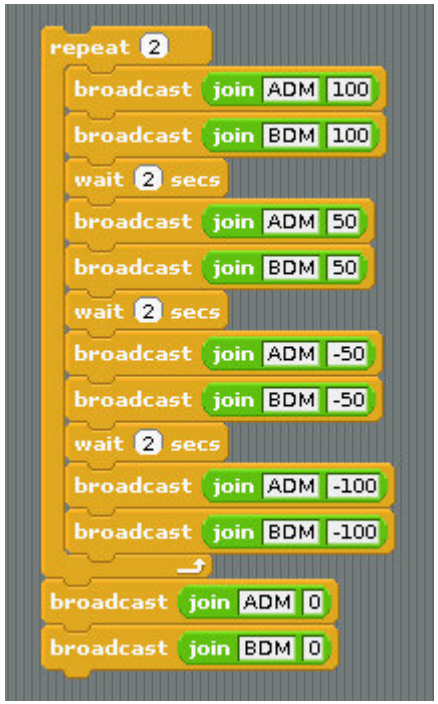
BDM50 DC Motor B speed 50

ADM-100 DC Motor A anticlockwise full speed 100

BDM-100 DC Motor B anticlockwise full speed 100

ADM0 DC Motor A stop

BDM0 DC Motor B stop



(2) BASIC GPIO Motor on off control

define GPIO 17,18,27,22, as output

1.press keyboard " f " GPIO 17,27 LED "ON" & 18,22 LED "OFF"

car move forward

2.press keyboard " b " GPIO 18,22 LED "ON" & 17,27 LED "OFF"

car move backward

3.press keyboard " l " GPIO 18,27 LED "ON" & 17,22 LED "OFF"

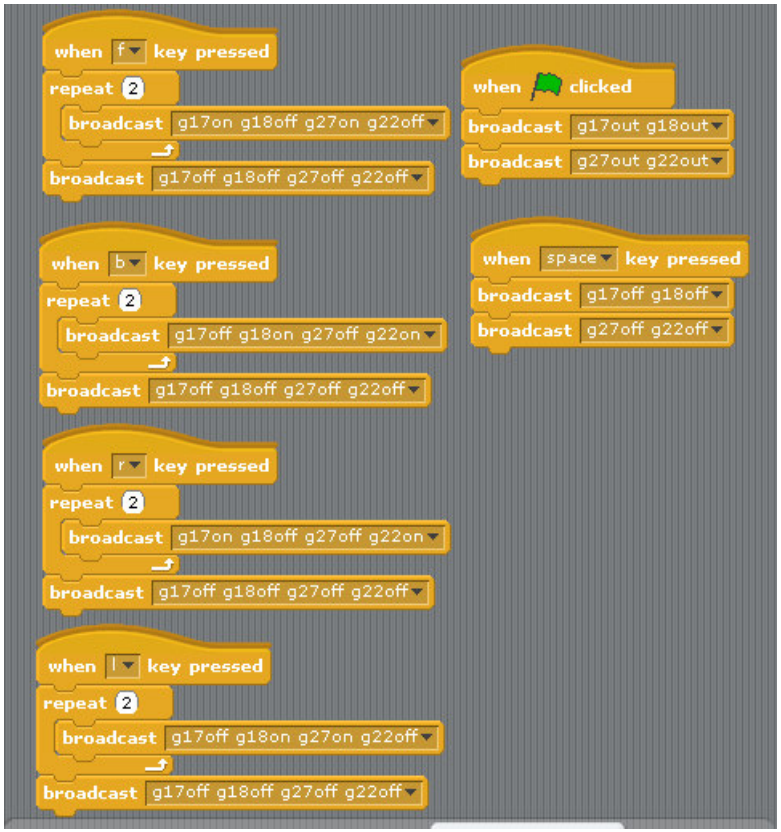
car turn right

4.press keyboard " r " GPIO 17,22 LED "ON" & 18,27 LED "OFF"

car turn left

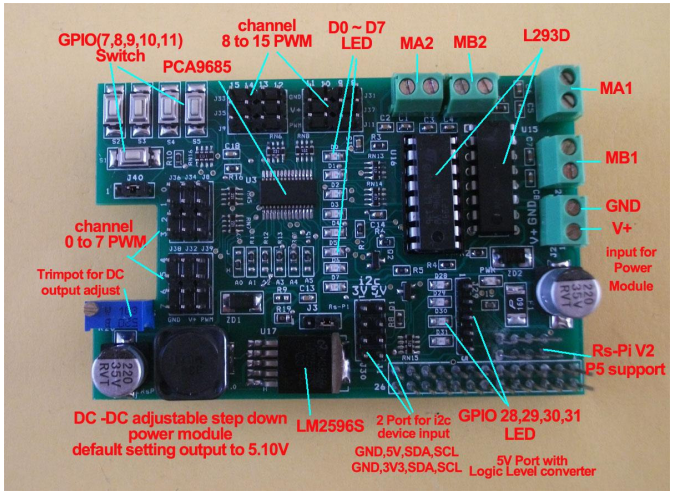
5.press keyboard " space " GPIO 17,27 LED "OFF" & 18,22 LED "OFF"

car STOP

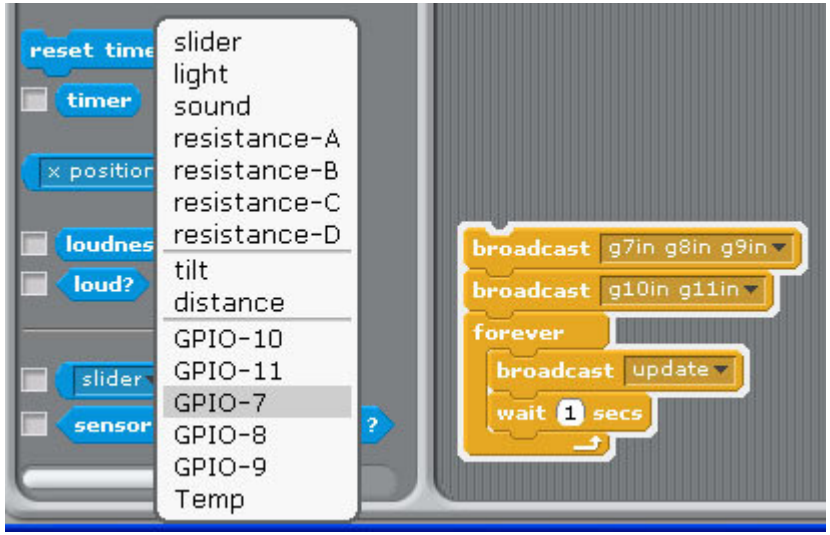


2 Motor control demo code forward, backward, turn right, turn left. Stop

7. Demo 2 5 GPIO Switch & 16 PWM & 4 Motor Board demo



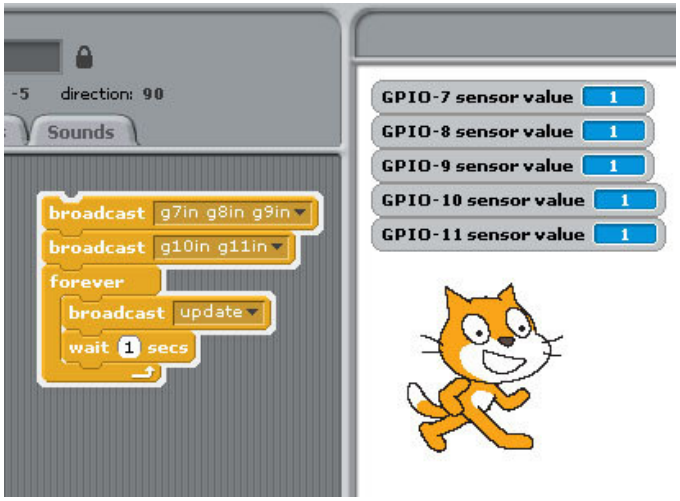
(1) 5 GPIO Switch GPIO 7,8,9,10,11 demo



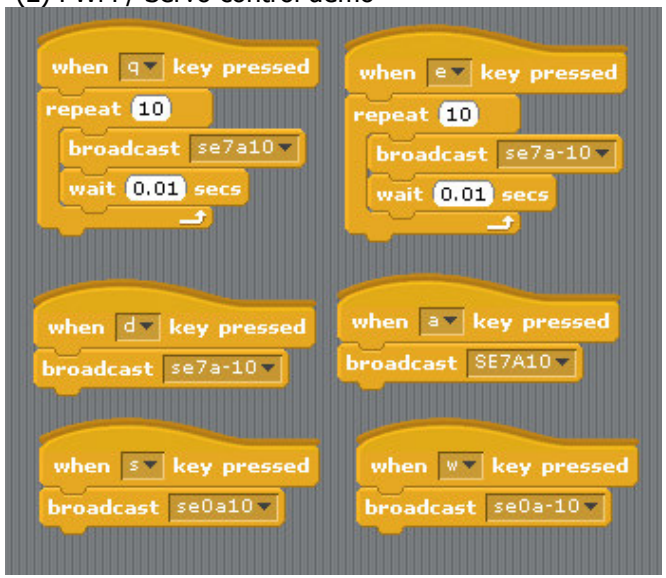
1.Setting GPIO 7,8,9,10,11 as input

2. broadcast "Update"

3. in Sensing --> Slider , you will see the GPIO-7 ,8, 9, 10, 11 in the list



(2) PWM / Servo control demo



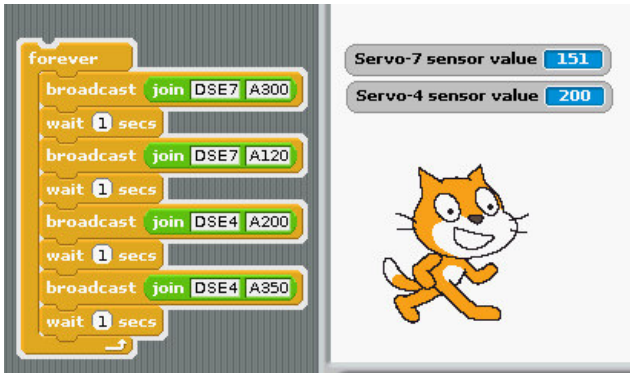

```
2 Servo in  channel 0 & channel 7
```

```
Command "SE"+ "PWM (0-15)" + "a" +"angle" Address 41
   se7a10   --> channel 7 servo move 10 angle address 41
   se7a-10  --> channel 7 servo move -10 angle address 41
   se0a10   --> channel 0 servo move 10 angle  address 41
   se0a-10  --> channel 0 servo move -10 angle  address 41
```


(3) Degree command

```
Command "DSE"+ "PWM (0-15)" + "a" +"degree" Address 41

   dse7a300  --> channel 7 servo move 10 angle address 41
   dse7a120  --> channel 7 servo move -10 angle address 41
   dse4a200  --> channel 0 servo move 10 angle  address 41
   dse4a350  --> channel 0 servo move -10 angle  address 41
```

after command " DSE7A300 " ==> channel 7 Servo to 300 degree
command " DSE4A200 " ==> channel 4 Servo to 200 degree
in Sensing --> Slider , you will see the " Servo-4, & Servo-7" in the list



8. Demo3 4hub/7hub i2c function board demo

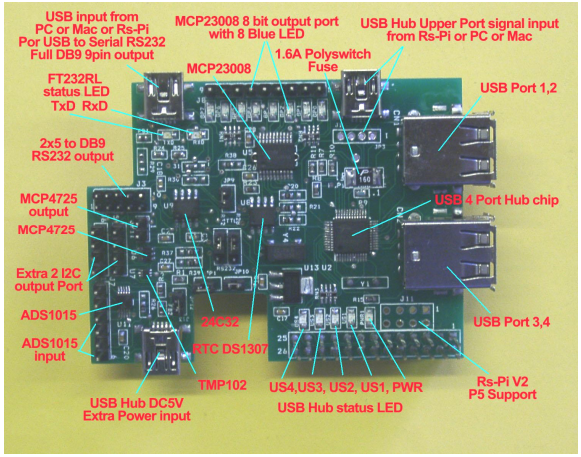
detect Temperature (i2c interface TMP102 - Temperature sensor) i2c ds1307 RTC , i2c 23008 8 GPIO , i2c 23c32 EEPROM, i2c adc ads1015 12bit Analog-to-Digital Converter

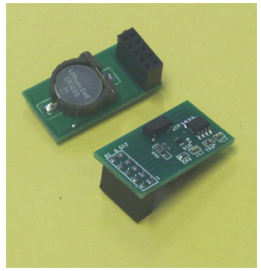
use our 4 USB Hub board with I2C RTC & TMP102 temperature sensor or RTC-Temp module board

20 -> MCP23008 48 -> TMP102 68 -> RTC DS1307

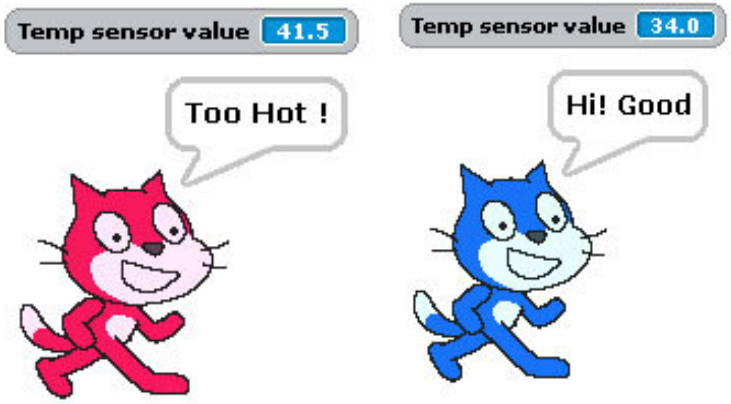
50 -> 24c32 60 -> MCP4725 49-> ADS1015

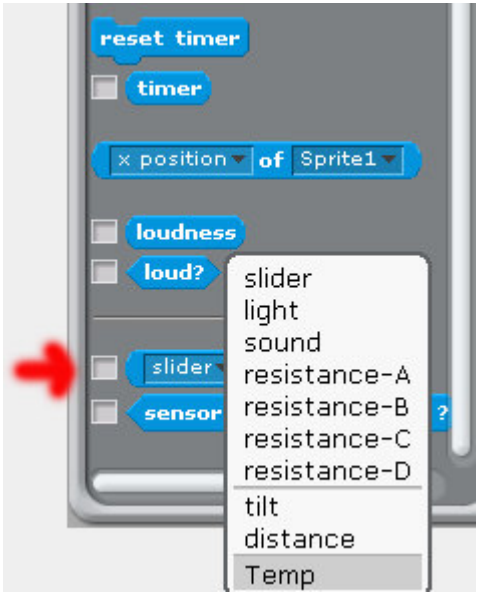
```
root@raspberrypi:~# i2cdetect -y 1
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:          -- -- -- -- -- -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: 20 -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- 48 49 -- -- -- -- -- --
50: 50 -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: 60 -- -- -- -- -- -- -- UU -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
root@raspberrypi:~#
```





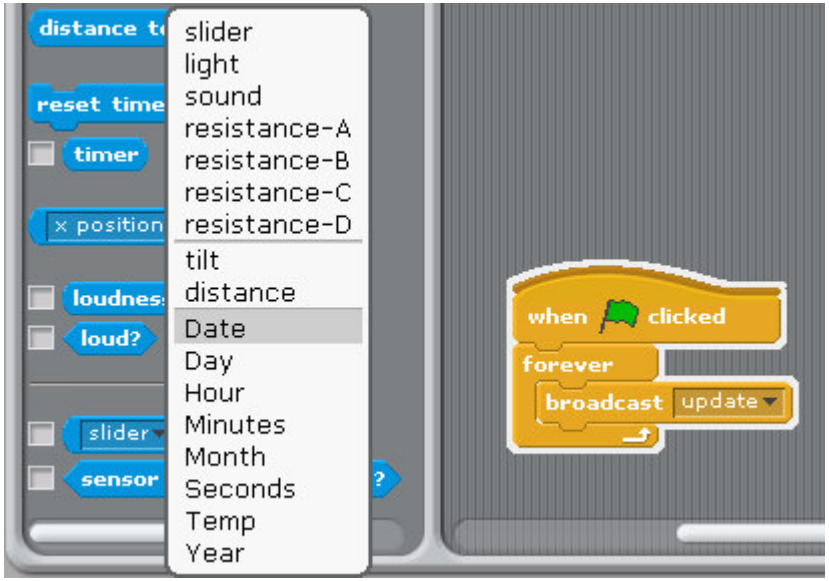
i2c address 48 TMP102 temperature sensor test





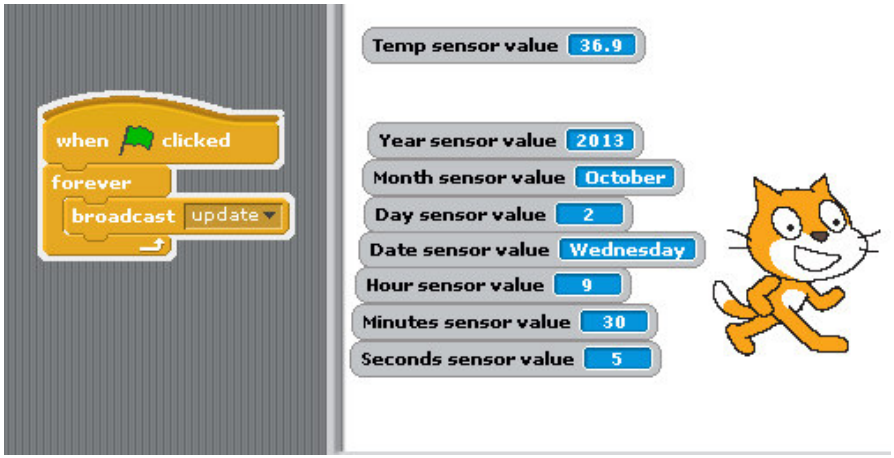
after broadcast "Update" in Sensing --> Slider , you will see the Temp in the list

(2) i2c address 68 DS1307 RTC detect & demo



after broadcast "Update" in Sensing --> Slider , you will see the RTC in the list

"Year, Month, Day, Date, Hour, Minutes, seconds"



RTC information from Raspberry Pi i2c RTC module

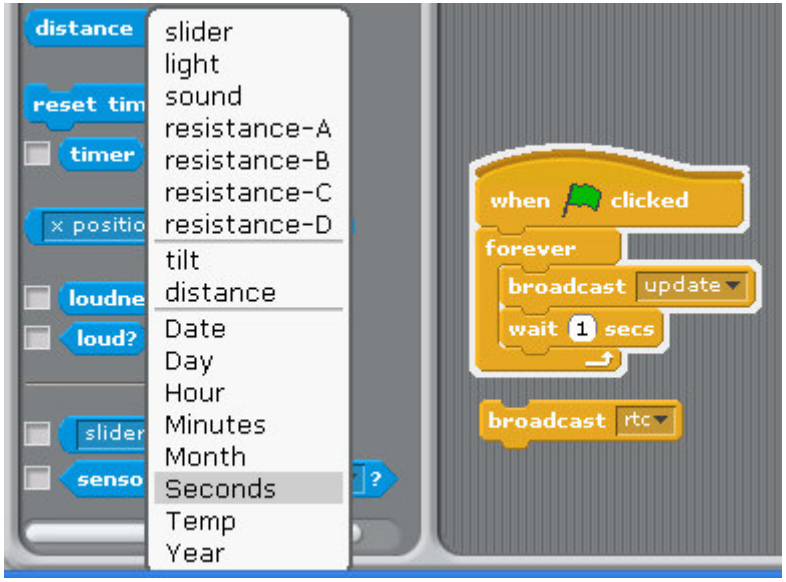
if you can't detect RTC function in Scratch

```
  GNU nano 2.2.6              File: /etc/rc.local

#
# By default this script does nothing.

# Print the IP address
_IP=$(hostname -I) || true
if [ "$_IP" ]; then
  printf "My IP address is %s\n" "$_IP"
fi

echo ds1307 0x68 > /sys/class/i2c-adapter/i2c-1/new_device

exit 0
```

you have echo ds1307 in /etc/rc.local

```
root@raspberrypi:~# i2cdetect -y 1
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:          -- -- -- -- -- -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: 20 -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- 48 49 -- -- -- -- -- --
50: 50 -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: 60 -- -- -- -- -- -- -- UU -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
root@raspberrypi:~#
```

in 0x68 address display "UU"

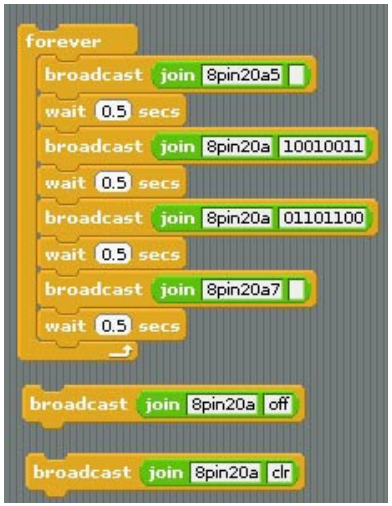


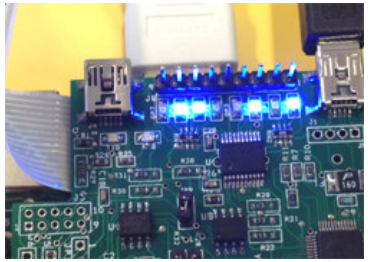
send broadcast "rtc" then broadcast "update"

after broadcast "Update" in Sensing --> Slider , you will see the RTC in the list

"Year, Month, Day, Date, Hour, Minutes, seconds"

(3) i2c address 20 23008 8 GPIO test





use command 8pin+"address"+a"bit (1 to 8)"

8pin+"address"+ a"10001000" --> 8bit mode

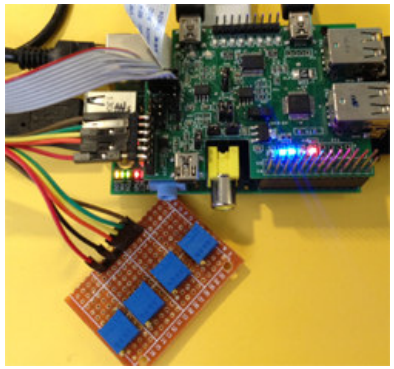
8pin20a5 --> output to address 20 bit 5

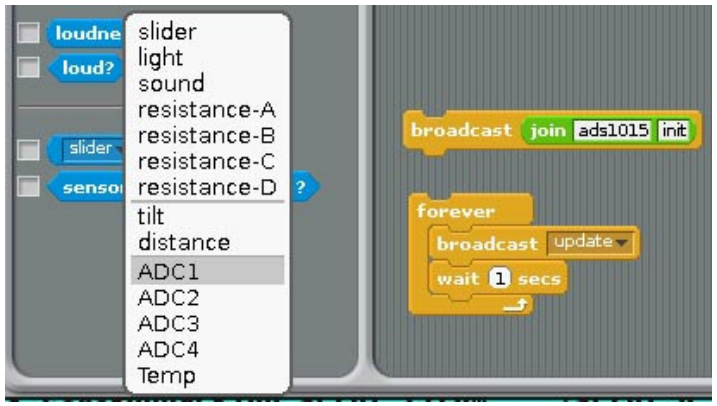
or use "8pin20a10010011" --> 8bit address 20 from 8 to 1 "10010011"

"8pin20aoff" or "8pin20aclr" all Port A 8bit off

(4) i2c address 49 adc ads1015 4 channel input test

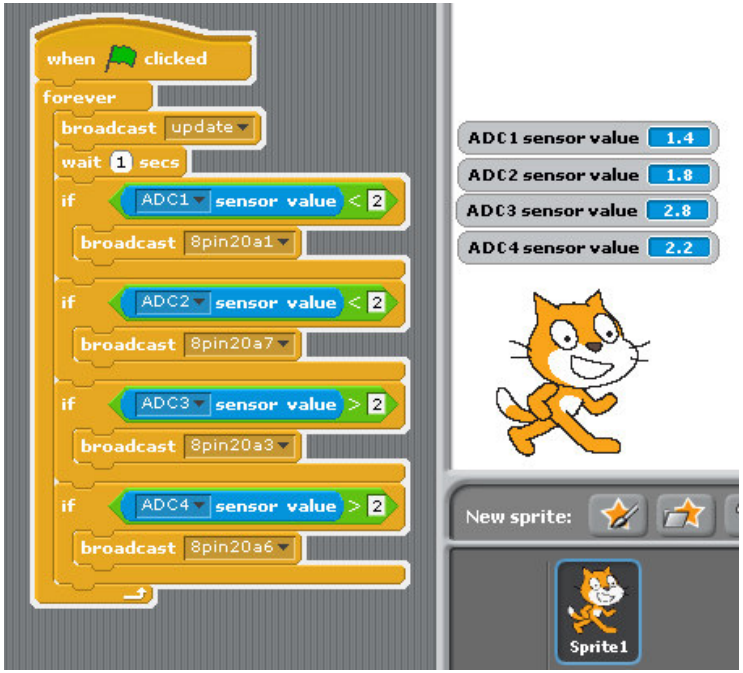
i2c adc ads1015 12bit Analog-to-Digital Converter





1. command "ads1015init" initial ADS1015

2.after broadcast "Update" in Sensing --> Slider , you will see the ADC1, ADC2,ADC3, ADC4 in the list



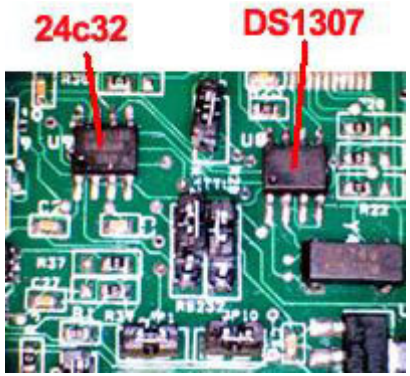
(1)adc1 input & output to 23008 8 bit GPIO bit 1

if adc1 < 2 23008 bit 1 LED "ON" 8pin20a1

(2) adc2 input & output to 23008 8 bit GPIO bit 7

if adc2 < 2 23008 bit 7 LED "ON" 8pin20a7

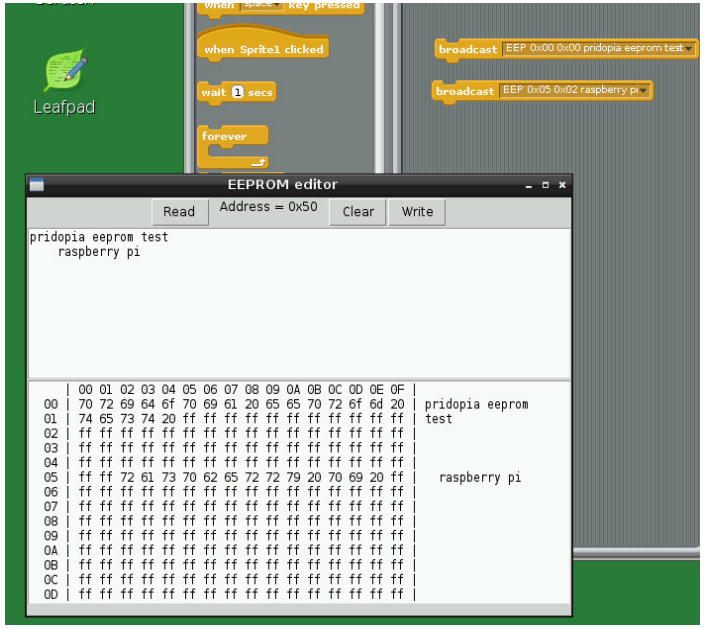
(5) EEPROM i2c address 50 24c32 read/write



command "EEP" + start address x, start address y + " message"

--> EEP 0x00 0x00 pridopia eeprom test

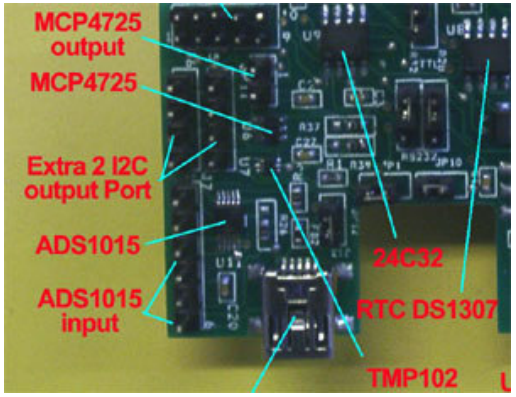
The System will open a new screen for EEPROM editor , need waiting about 10 seconds

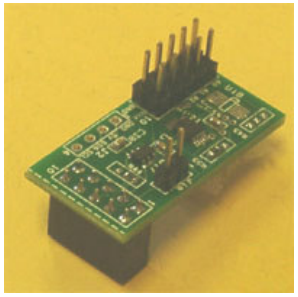


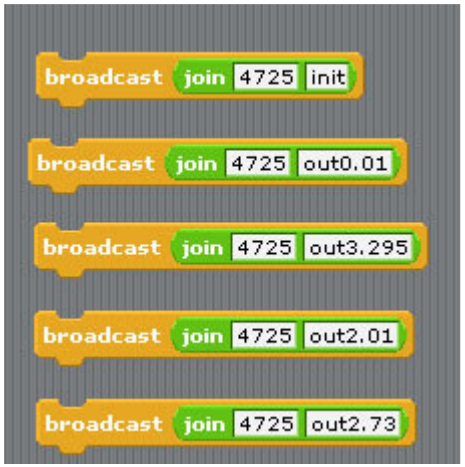
(6) DAC MCP4725 12bit Digital-to-Analog Converter

JP11 (AOUT, GND) i2c address 60

demo our 4Hub/7Hub AD/DA board , AD/DA module board

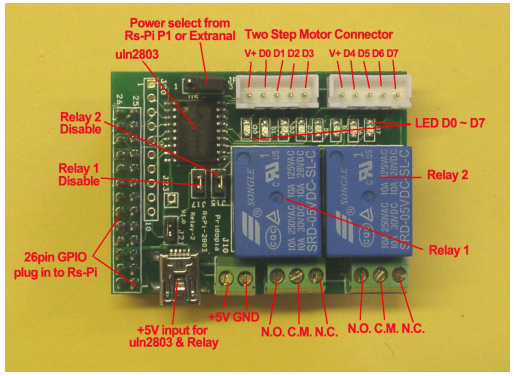


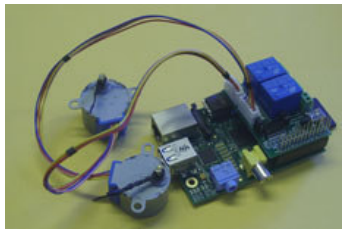





```
Command "4725init" initial  DAC mcp4725
Command "4725out" + "voltage"
 voltage range "DC 0.01V to DC 3.29V"
command "4725out2.73"   DC output 2.73V
```

9 Demo4 Stepper Motor and Relay control

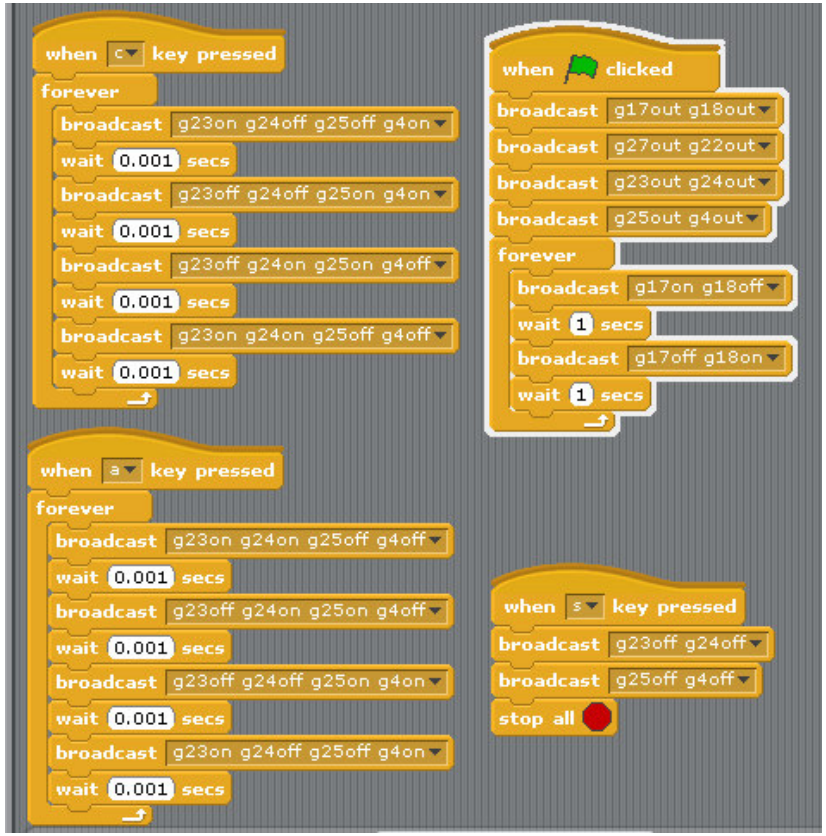




main program turn relay 1,on 2 off -> 1 off, 2 on on cycle

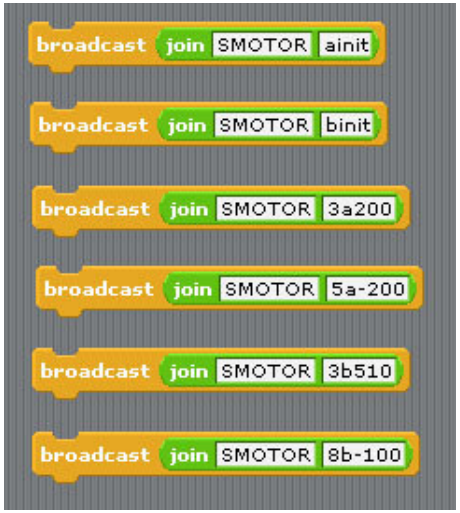
press C Stepper Motor clockwise press A Stepper Motor anticlockwise

Press S Stop



(2) Step Motor smotor command easy way

command "SMOTORainit" initial GPIO 17,18,27,22 as A Step Motor Port
command "SMOTORbinit" initial GPIO 23,24,25,7 as B Step Motor Port
command "SMOTOR" + speed (3-50) +(a or B) + output step
100 --- clockwise 100 step , -100 --- anticlockwise 100 step
a -- GPIO 17,18,27,22 , b -- GPIO 23,24,25,4



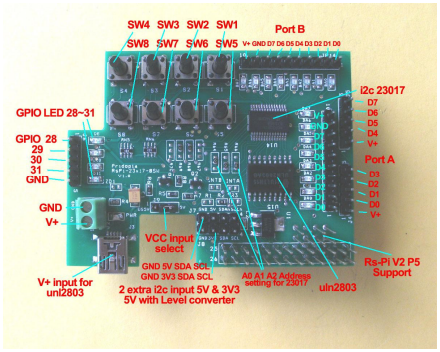
10. Demo5 Input & output test

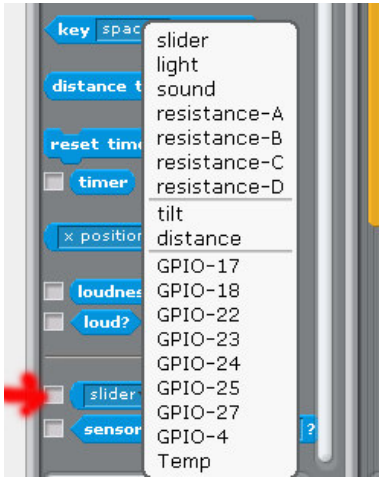
1.Setting GPIO (17,18,27,22,23,24,25,4)

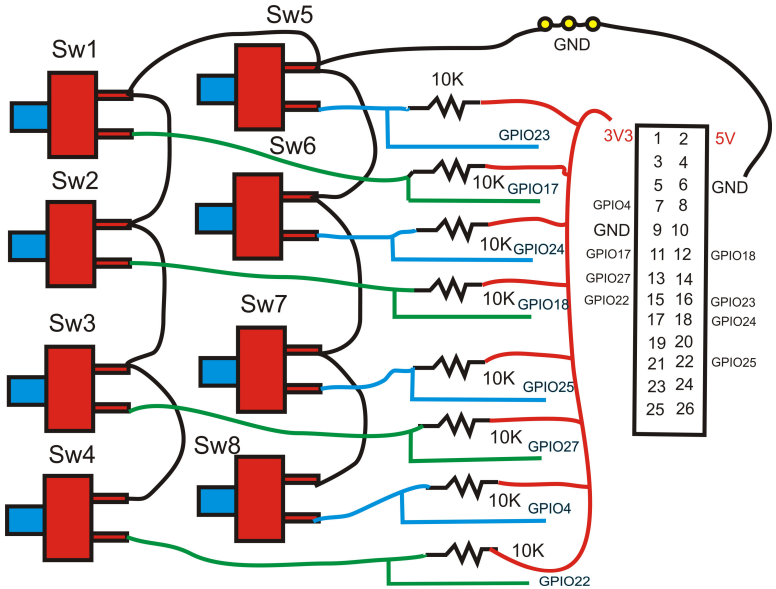
as input control Switch 1 to 8 ,

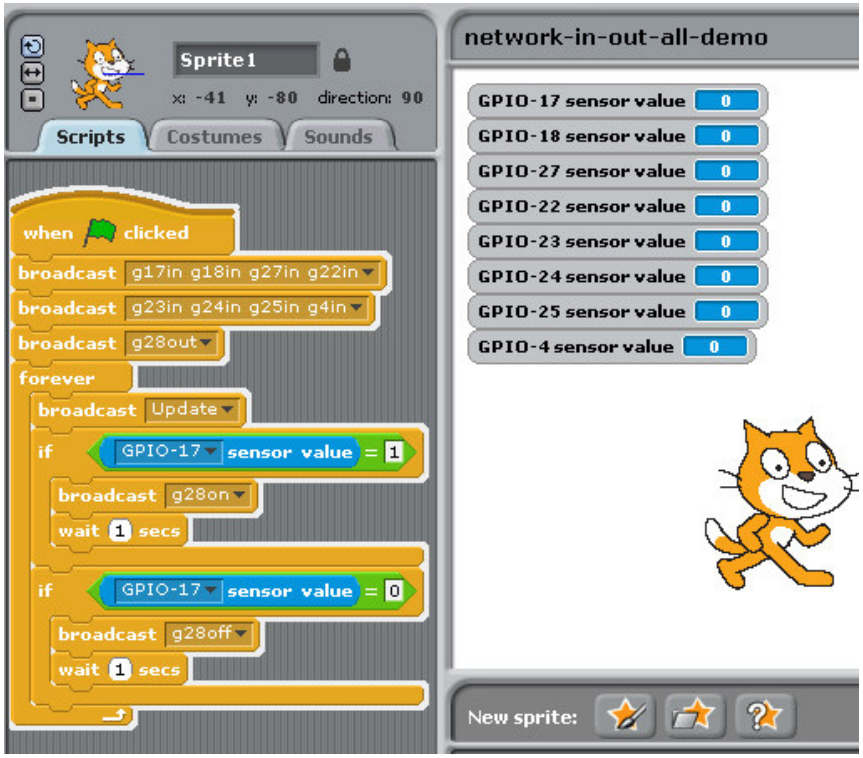
2. broadcast "Update"

3. in Sensing --> Slider , you will see the GPIO 17,18,27,22,23,24,25,4 in the list









Detect GPIO (17,18,27,22,23,24,25,4) switch status

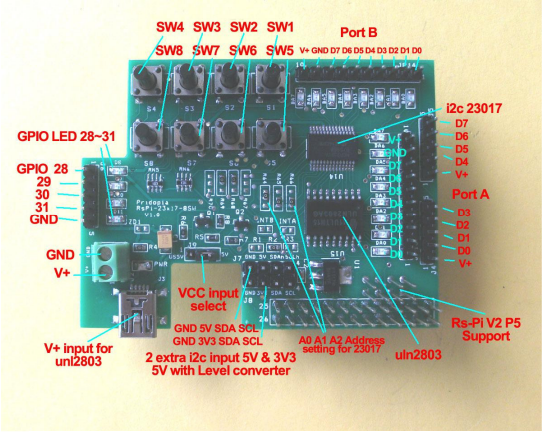
Press GPIO17 input switch output to GPIO28 LED

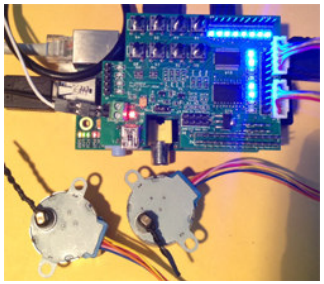
11. Demo6 Input & i2c 23017 16 GPIO & 16 LED output test & Stepper Motor Test

1.Setting GPIO 17 as input control Switch 1

2. broadcast "Update"

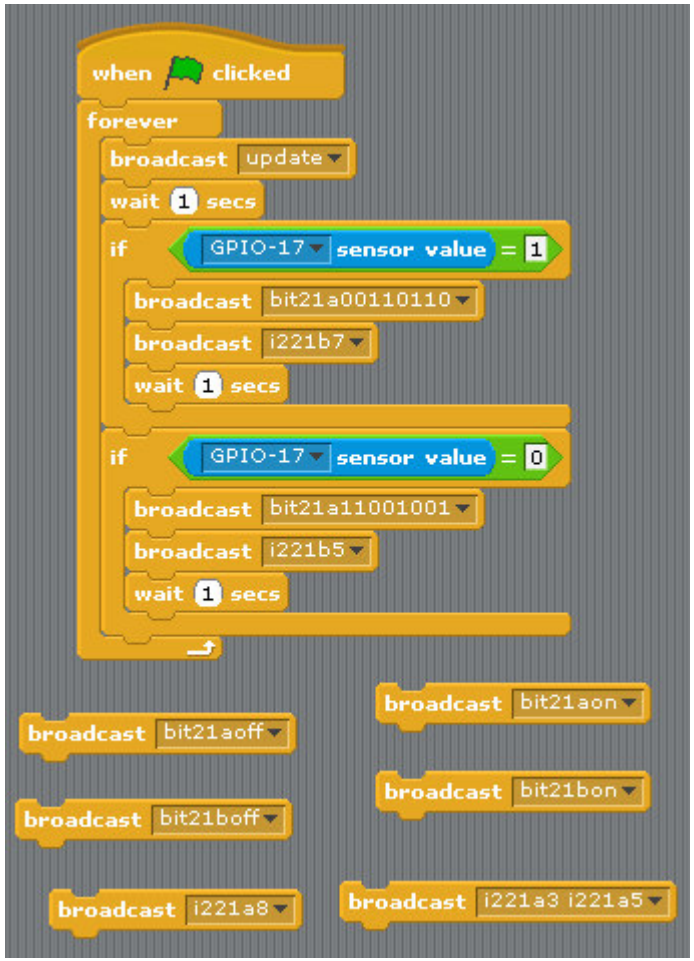
3. in Sensing --> Slider , you will see the GPIO 17 in the list



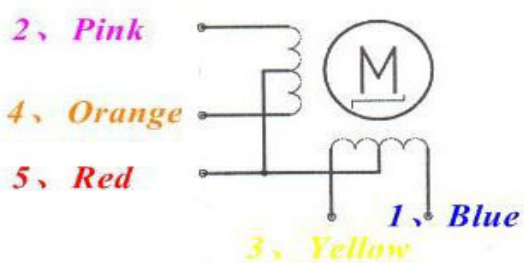


(1) switch input & i2c 16 GPIO output

```
command "bit21aon"   or "bit21bon"
i2c 23017 address 21 PortA ,B  16LED all on
command "bit21aoff"   or "bit21boff"
i2c 23017 address 21 Port A/B  16LED all off
or use "bit21a01100000" address 21 Port A bit 7,6  on
   use "bit21a10011111" address 21 Port A bit
8,5,4,3,2,1  on
"i221a8"  i2c address 21 port A bit 8  on
"i221a5"  i2c address 21 port A bit 5  on
"i221b3"  i2c address 21 port B bit 3  on
"i221b8"  i2c address 21 port B bit 8  on
```



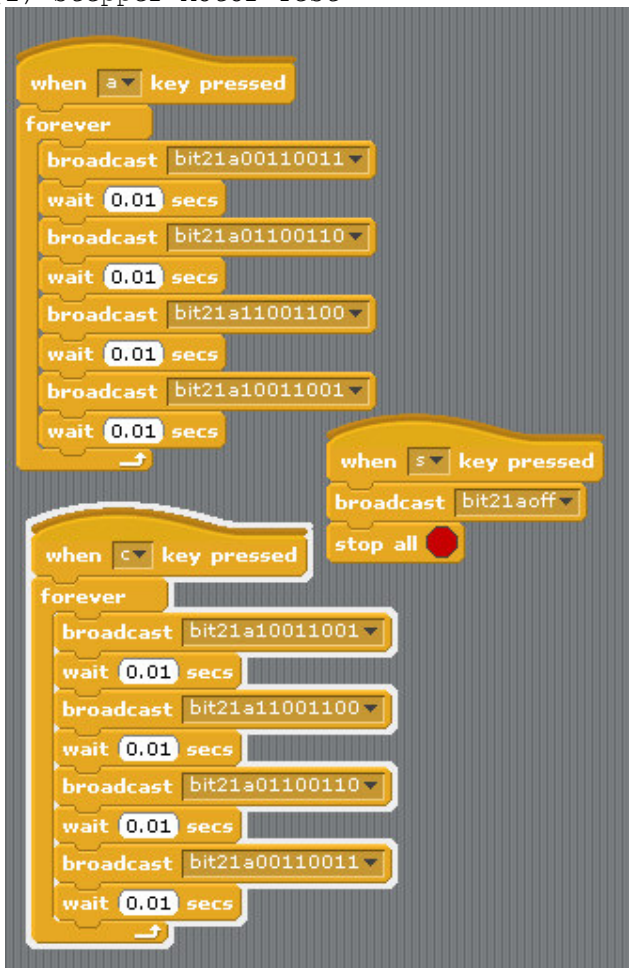
set GPIO17 as input, output to i2c Port A/B



5V Step Motor

Lead Wire Color	---> CW Direction (1-2 Phase)							
	1	2	3	4	5	6	7	8
4 ORG	-	-						-
3 YEL		-	-	-				
2 PIK				-	-	-		
1 BLU						-	-	-

(2) Stepper Motor Test



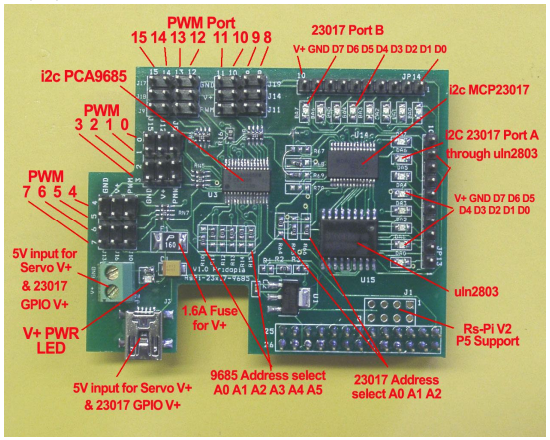
press C address 21 GPIO Stepper Motor clockwise

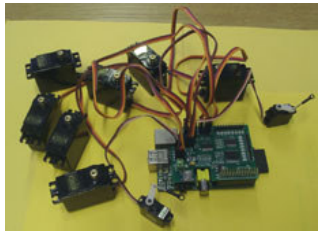
press A address 21 GPIO Stepper Motor anticlockwise

Press S Stop

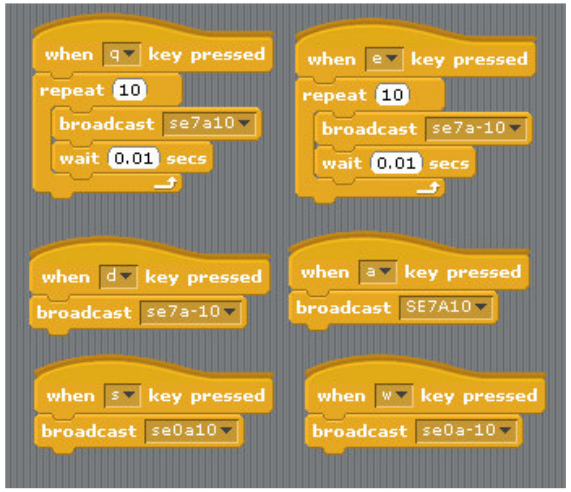
12. Demo 7 i2c 16 channel PWM & i2c 23017 16 GPIO output test

(1) Servo Motor control



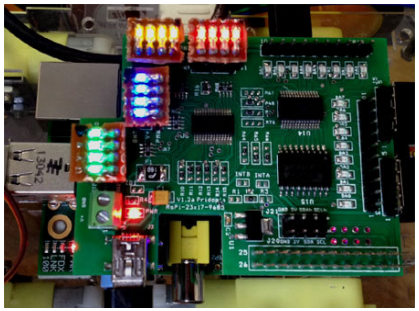


```
Command "SE"+ "PWM (0-15)" + "a" +"angle"  for Address 41
   se7a10   --> channel 7 servo move 10 angle  address 41
   se7a-10  --> channel 7 servo move -10 angle address 41
   se0a10   --> channel 0 servo move 10 angle  address 41
   se0a-10  --> channel 0 servo move -10 angle  address 41
```

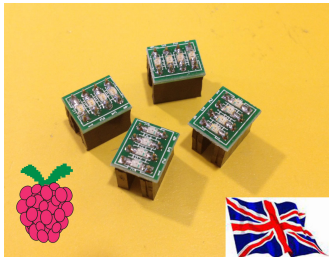


2 Servo in channel 0 & channel 7

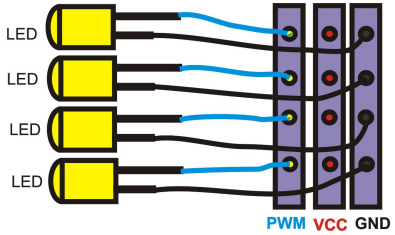
(2) PWM LED control 16 channel LED control

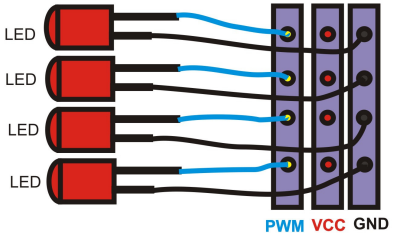


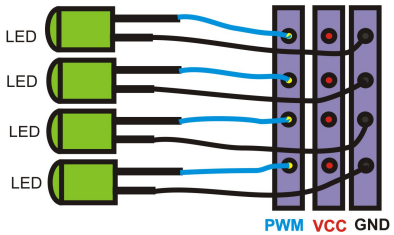
16 channel PWM board

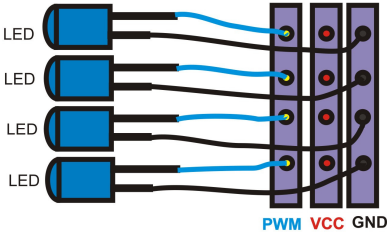


LED module (Blue, Green,Yellow,Red)









LED Scrolling Command PWMLED41S0E15D4True

PWMLED"Address" S[Start channel] E[End channel] D[Delay / Timing] [True/False]

Address 41, 42,43,44

Start channel & **E**nd channel 0 ~ 15 16 channel

D 1,2,3,4,5 (1 ~ 100) Delay Timing

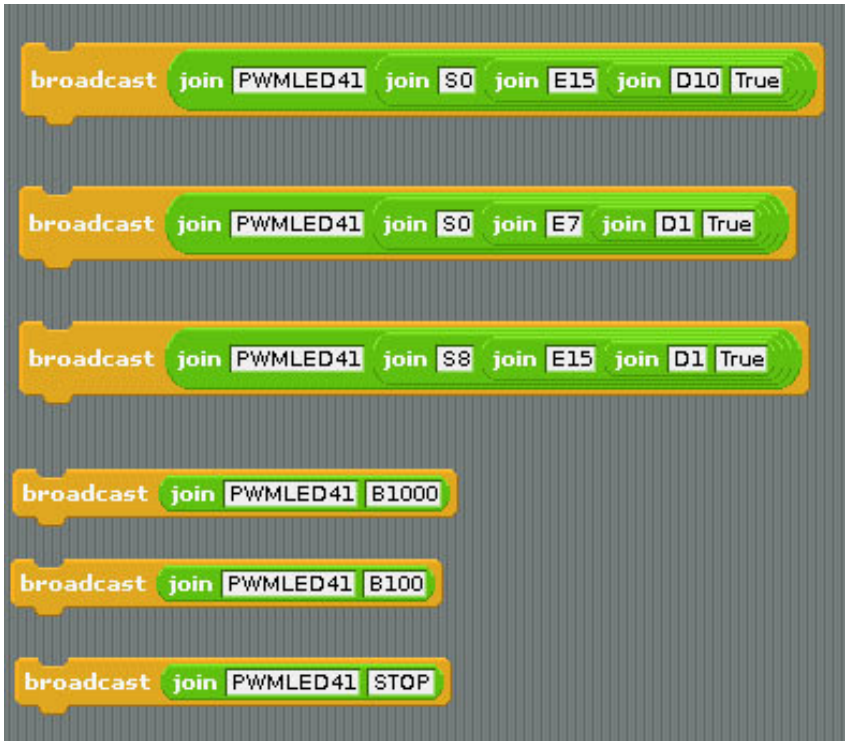
LED Brightness control Command

PWMLED "Address" "B" "0 ~ 1000"

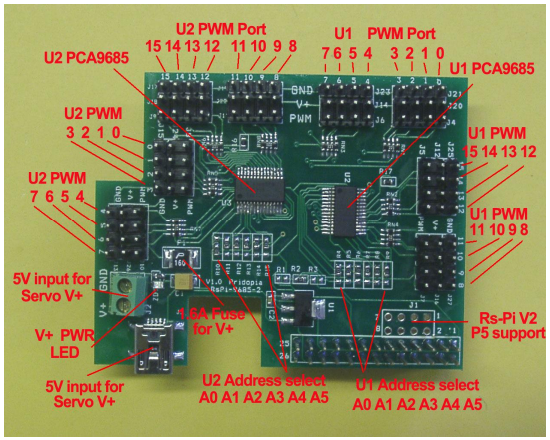
PWMLED41B1000

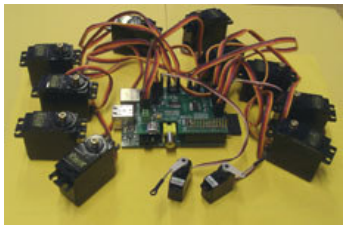
Stop command

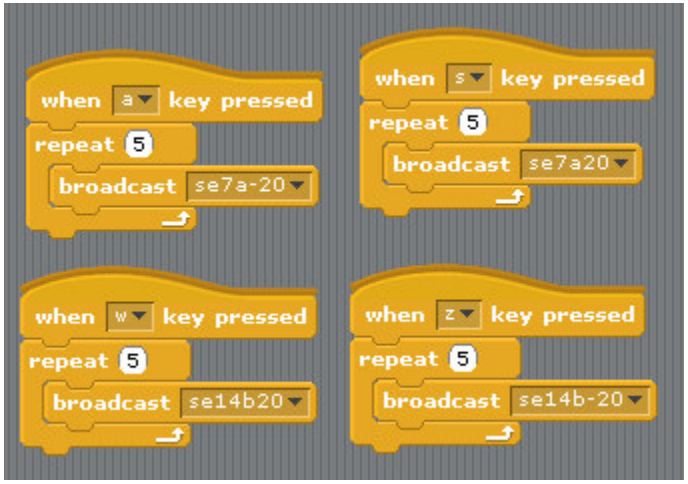
"PWMLED41STOP"



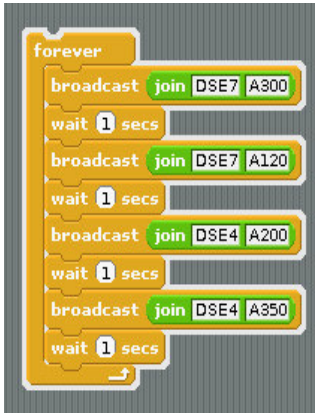
13. Demo 8 i2c 32 channel PWM Test

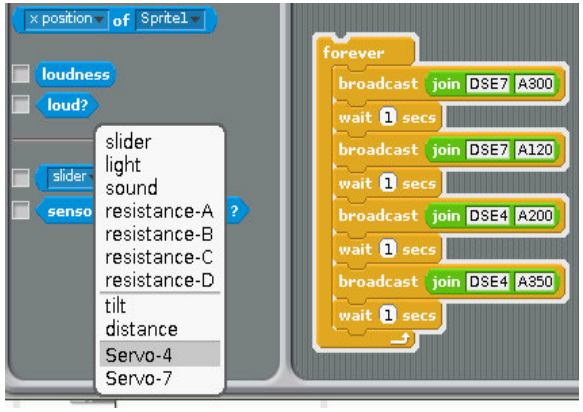






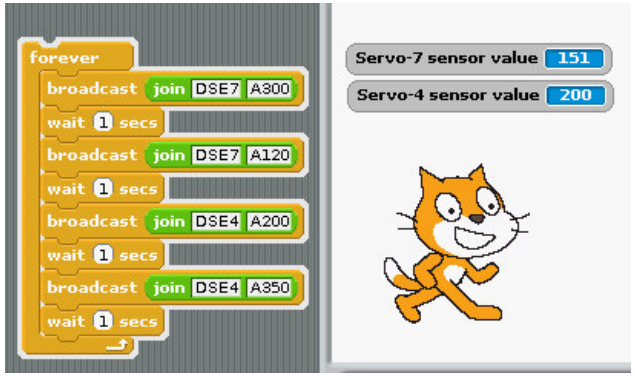

```
Command "SE"+ "PWM (0-15)" + "a" +"angle"  Address 41
Command "SE"+ "PWM (0-15)" + "b" +"angle"    for Address 40
  se7a20      --> channel 7 servo move 20 angle   address 41
  se7a-20     --> channel 7 servo move -20 angle  address 41
  se14b20     --> channel 14 servo move 20 angle  address 40
  se14b-20    --> channel 14 servo move -20 angle address 40
```



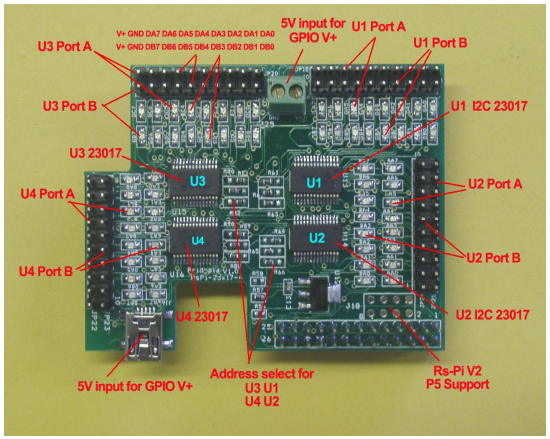


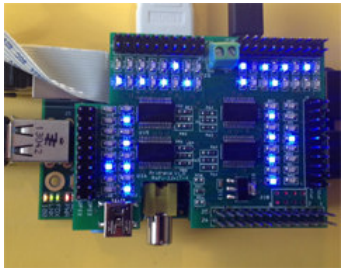

```
Command "DSE"+ "PWM (0-15)" + "a" +"degree" Address 41

   dse7a300  --> channel 7 servo move 10 angle address 41
    dse7a120  --> channel 7 servo move -10 angle address 41
    dse4a200  --> channel 0 servo move 10 angle  address 41
    dse4a350  --> channel 0 servo move -10 angle  address 41
```

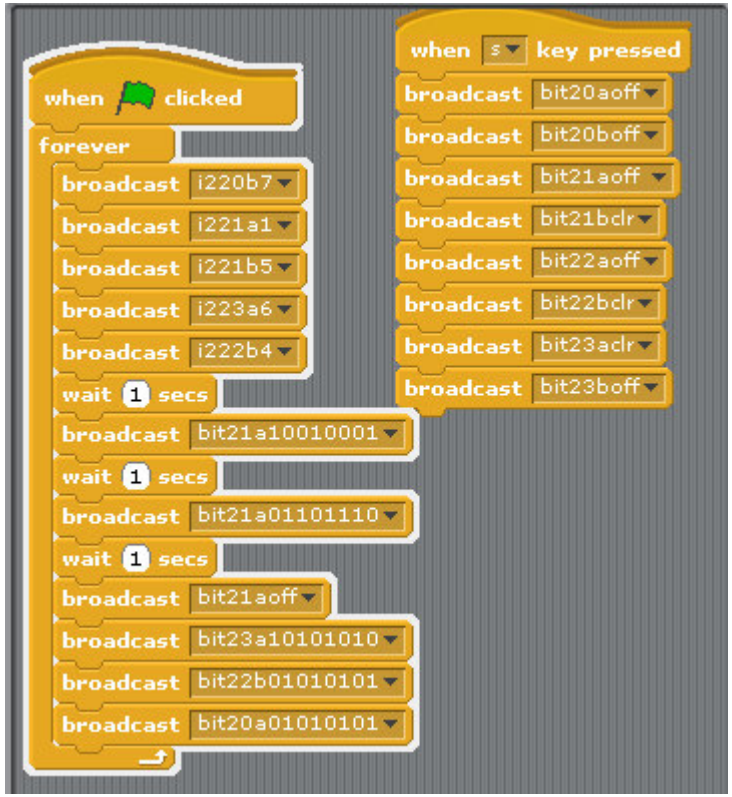
after command " DSE7A300 " ==> channel 7 Servo to 300 degree
command " DSE4A200 " ==> channel 4 Servo to 200 degree
in Sensing --> Slider , you will see the " Servo-4, & Servo-7" in the list



14. Demo 9 i2c 23017-4 64 GPIO & 64 LED output test



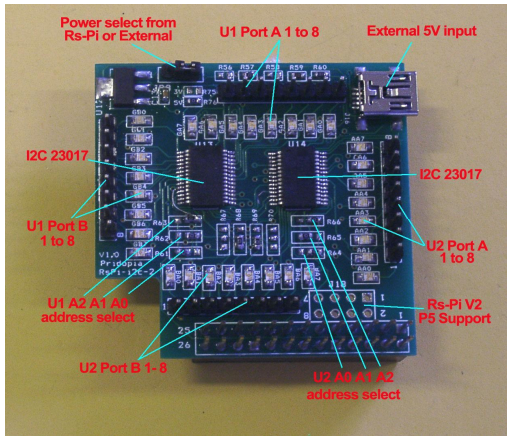


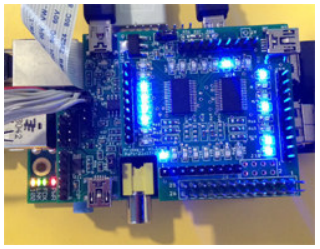


U1 to U4 i2c 23017 address 20,21,22,23

```
 Command "i2"+ "address(20-27)" + "a" +"bit(1 to 8)" for
Port A
 Command "i2"+ "address(20-27)" + "b" +"bit(1 to 8)" for
Port B
 Command "bit"+ "address(20-27)" + "a" +"bit(8 to 1)"
for Port A
 Command "bit"+ "address(20-27)" + "b" +"bit(8 to 1)"
for Port B

  i221a1 -->  i2c address 21 Port A bit 1 ON/OFF
  i220b7 -->  i2c address 20 Port B bit 7 ON/OFF
  i222b4 -->  i2c address 22 Port B bit 4 ON/OFF
  bit22b01010101 -->  address 22 port B from bit 8 to 1
                      output  --> 01010101
  bit20a01010101 -->  address 20 port A from bit 8 to 1
                      output  --> 01010101
  bit21aoff  --> address 21 Port A all OFF/clear
  bit23aclr  --> address 23 Port A all OFF/clear
```


15. Demo10 i2c 23017-2 32 GPIO & 32 LED output test



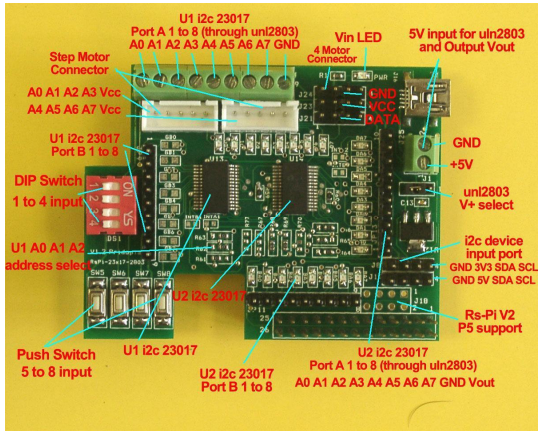


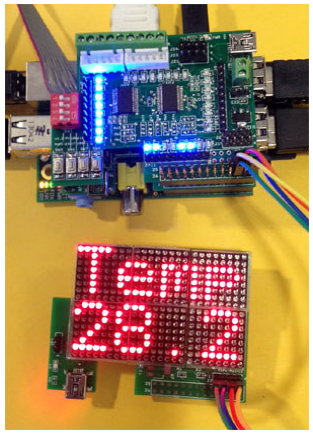
U1 to U2 i2c 23017 address 21,22

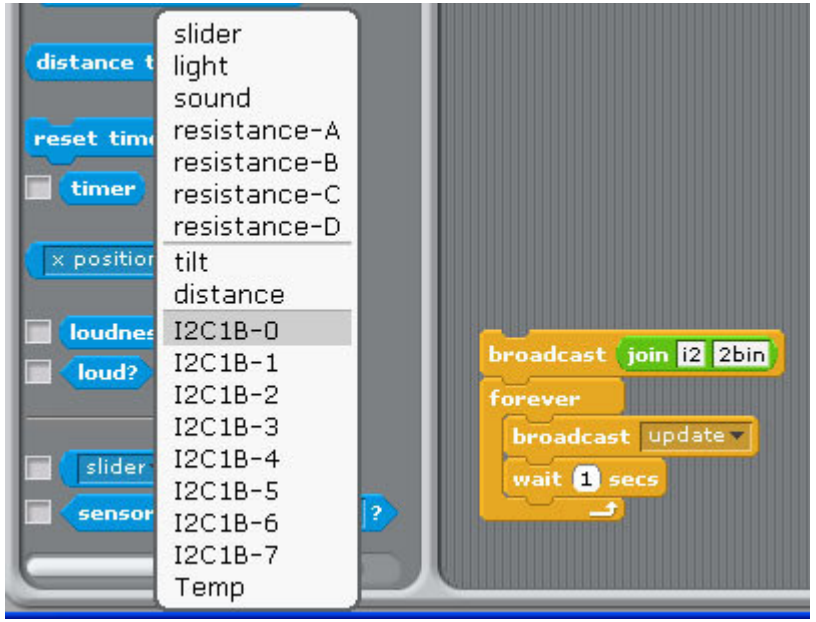
```
 Command "i2"+ "address(20-27)" + "a" +"bit(1 to 8)" for
Port A
 Command "i2"+ "address(20-27)" + "b" +"bit(1 to 8)" for
Port B
 Command "bit"+ "address(20-27)" + "a" +"bit(8 to 1)"
for Port A
 Command "bit"+ "address(20-27)" + "b" +"bit(8 to 1)"
for Port B

  i221a1 -->  i2c address 21 Port A bit 1 ON/OFF
  i222b4 -->  i2c address 22 Port B bit 4 ON/OFF
  bit22b01010101 -->  address 22 port B from bit 8 to 1
                      output  --> 01010101
  bit21a01010101 -->  address 21 port A from bit 8 to 1
                      output  --> 01010101
  bit21aoff  --> address 21 Port A all OFF/clear
  bit21boff  --> address 21 Port B all OFF/clear
  bit22aoff  --> address 22 Port A all OFF/clear
```


16. Demo 11 23017-2803 Switch input test

i2c 32GPIO & 8 switch input







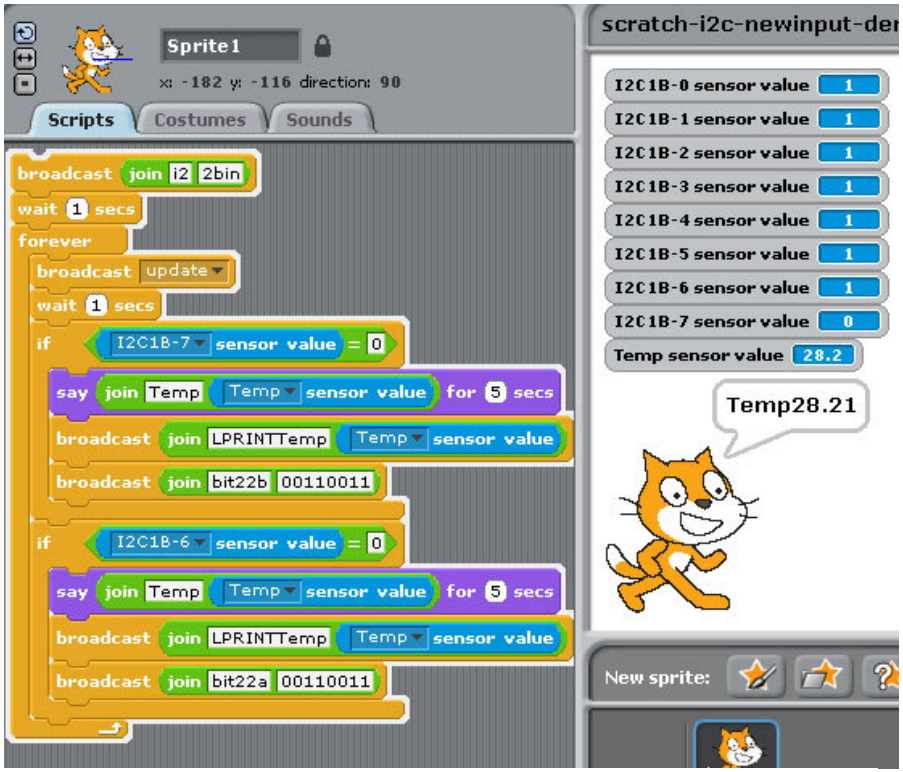

```
Command "i2"+ "address(1-8)" + "a" +"in" for Port A
Command "i2"+ "address(1-8)" + "b" +"in" for Port B
 Address 20 --> 1 21 --> 2 22-->3 23 -->4
 Address 24 --> 5 25 --> 6 26-->7 27 -->8
```

```
command "i22bin" initial address 21, Port B as input
```

```
(1)"i22bin" initial address 21, Port B as input
```

(2) broadcast "Update"

(3) in Sensing --> Slider , you will see the " I2C1B-0 ~ I2C1B-7" in the list



The demo use i2c 24x16 LED Matrix for output message

The demo use i2c 24x16 LED Matrix for output message

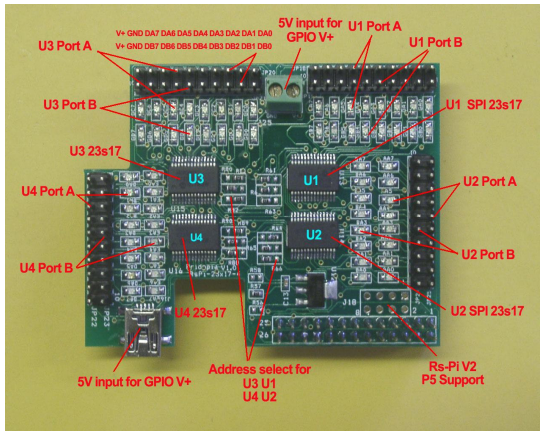
i2c TMP102 for temperature sensor

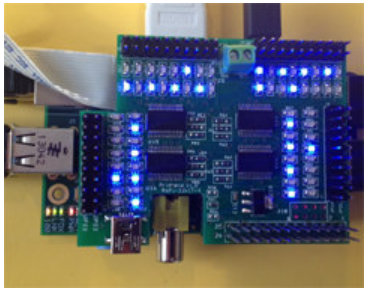
i2c 23017 address 21 port B for 8 switch input

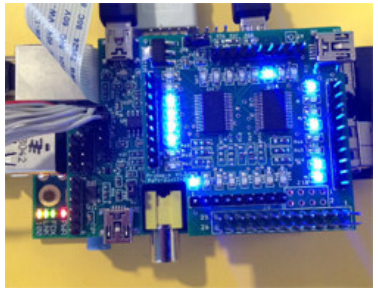
i2c 23017 address 22 port A/B for output

17. Demo12 23S17 32 /64 SPI GPIO Board

spi 23s17-4 64 GPIO & 64 LED output test
spi 23s17-2 32 GPIO & 32 LED output test

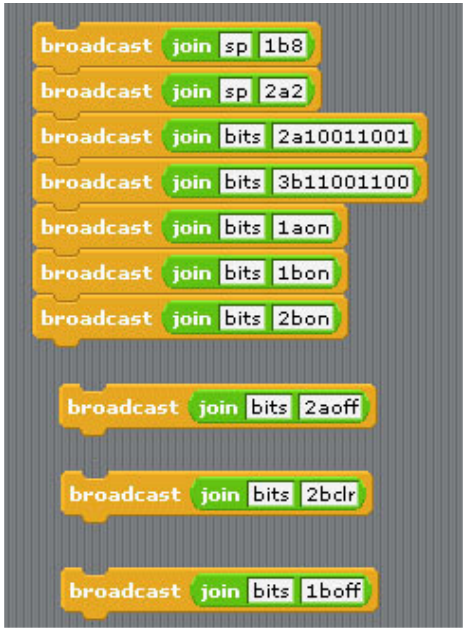




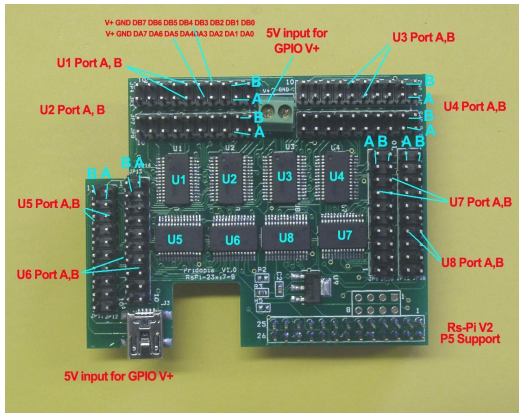


```
U1 to U4 spi 23s17  address 40,42,44,46
U1 to U2 spi 23s17  address 40,42

40 --> 1  42 --> 2  44 --> 3  46  --> 4
48 --> 5  4a --> 6  4c --> 7  4e  --> 8

 Command "sp"+ "address(1-8)" + "a" +"bit(1 to 8)" for
Port A
 Command "sp"+ "address(1-8)" + "b" +"bit(1 to 8)" for
Port B
 Command "bits"+ "address(1-8)" + "a" +"bit(8 to 1)" for
Port A
 Command "bits"+ "address(1-8)" + "b" +"bit(8 to 1)" for
Port B
```

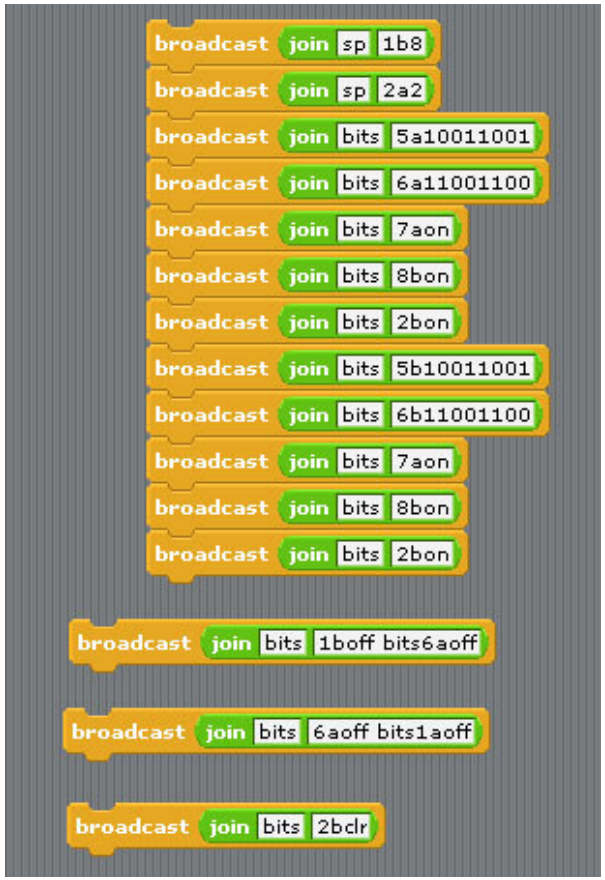
```
sp1a1 -->  spi address 1 Port A bit 1 ON/OFF
sp2b7 -->  spi address 2 Port B bit 7 ON/OFF
sp3b4 -->  spi address 3 Port B bit 4 ON/OFF

bits2b01010101 -->  address 2 port B from bit 8 to 1
            output  --> 01010101
bits2a01010101 -->  address 2 port A from bit 8 to 1
                    output  --> 01010101
bits2aoff  --> address 2 Port A all OFF/clear
bits2aclr  --> address 2 Port A all OFF/clear
```



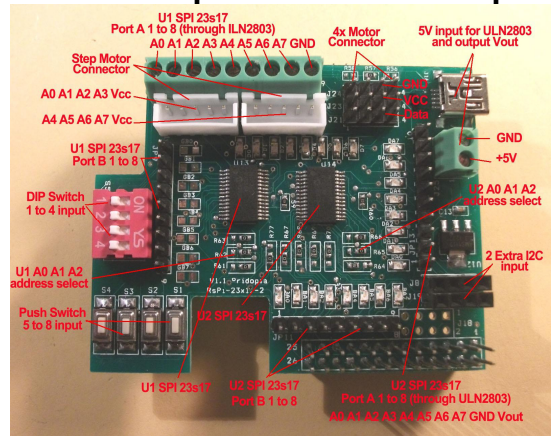
18. Demo13 spi 23s17-8 128 GPIO



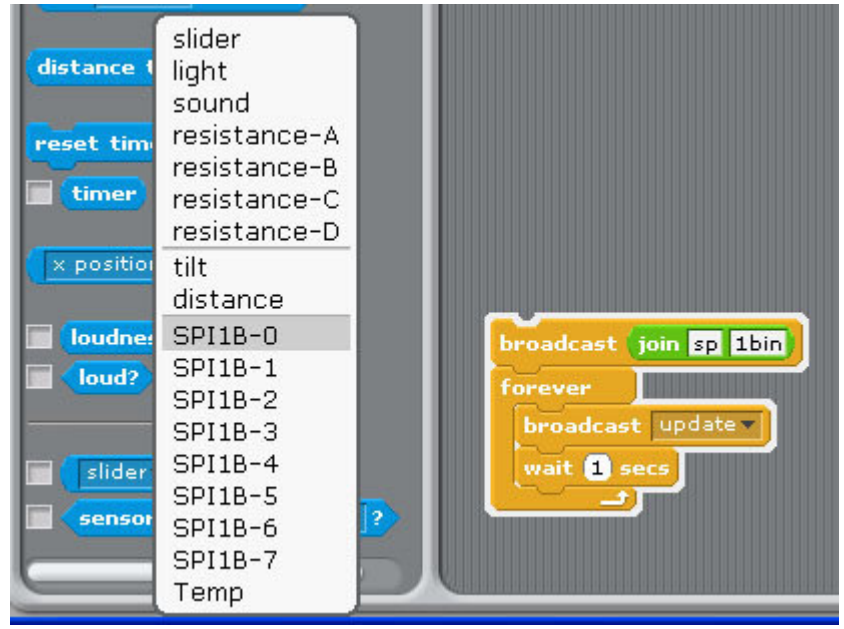

```
U1 to U4 spi 23s17  address 40,42,44,46
U5 to U8 spi 23s17  address 48,4a,4c,4e

40 --> 1  42 --> 2  44 --> 3  46  --> 4
48 --> 5  4a --> 6  4c --> 7  4e  --> 8

 Command "sp"+ "address(1-8)" + "a" +"bit(1 to 8)" for
Port A
 Command "sp"+ "address(1-8)" + "b" +"bit(1 to 8)" for Port B
 Command "bits"+ "address(1-8)" + "a" +"bit(8 to 1)" for Port A
 Command "bits"+ "address(1-8)" + "b" +"bit(8 to 1)" for Port B
  sp1a1 -->  spi address 1 Port A bit 1 ON/OFF
  sp5b7 -->  spi address 5 Port B bit 7 ON/OFF
  sp7b4 -->  spi address 7 Port B bit 4 ON/OFF
  bits2b01010101 -->  address 2 port B from bit 8 to 1
                      output  --> 01010101
  bits8a01010101 -->  address 8 port A from bit 8 to 1
                      output  --> 01010101
  bits2aoff  --> address 2 Port A all OFF/clear
  bits2aclr  --> address 2 Port A all OFF/clear
```



19. Demo 14 spi 23s17-2 8 Switch input



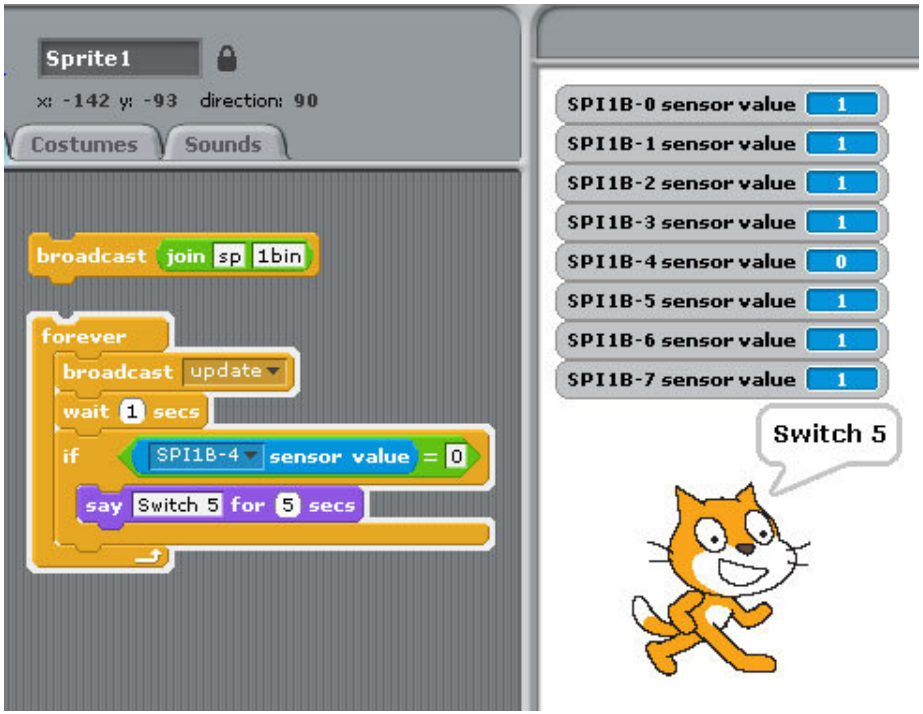
U1 to U2 spi 23s17 address 40,42,44,46



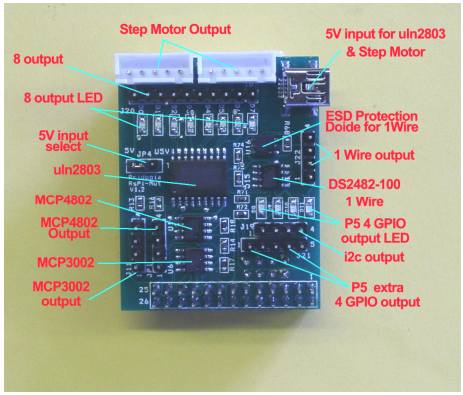
Command "sp"+ "address(1-8)" + "a" +"in" for Port A
Command "sp"+ "address(1-8)" + "b" +"in" for Port B

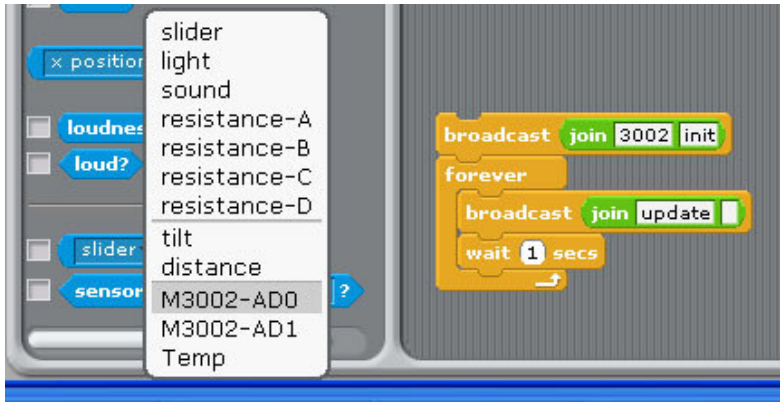
command "sp1bin" initial address 40, Port B as input
broadcast "Update" in Sensing ---> Slider ,

you will see "SPI1B-0 ~ SPI1B-7" in the list

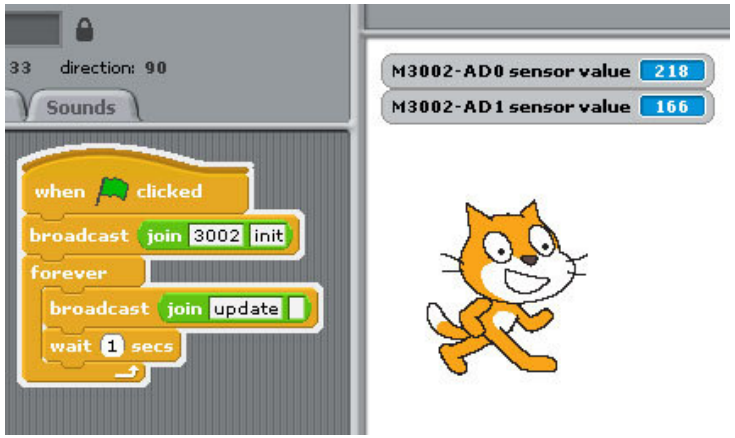


20. Demo 15 SPI MCP3002 & MCP4802 AD/DA demo

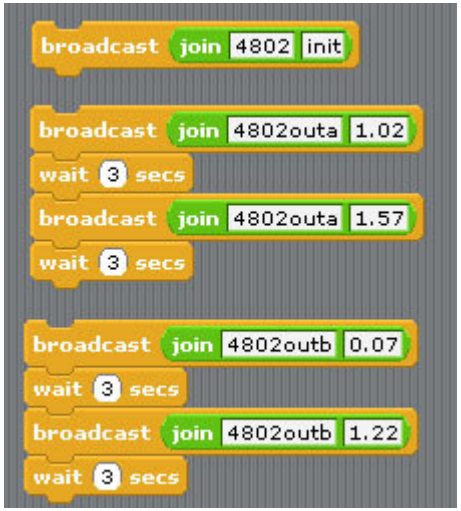




command "3002init" initial spi mcp3002
broadcast "Update" in Sensing ---> Slider ,
you will see "M3002-AD0, M3002AD1" in the list

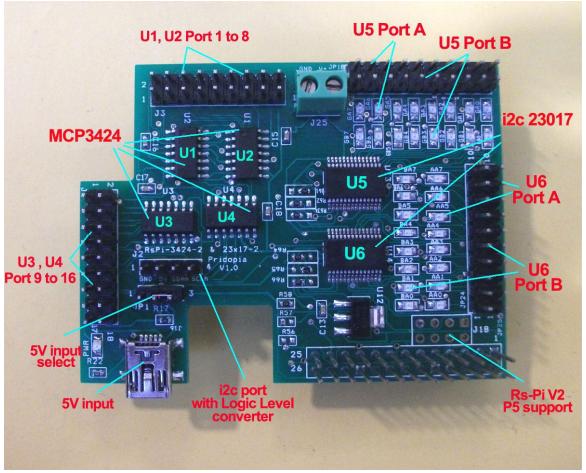


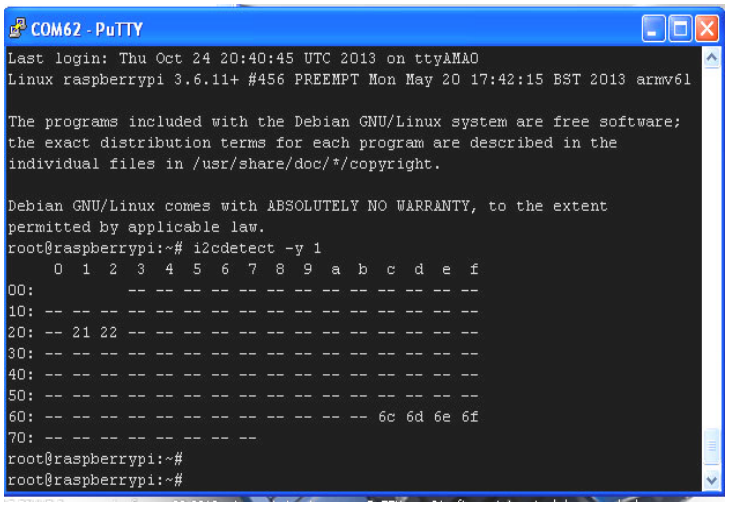
(2)MCP4802 DAC 8 bit Digital-to-Analog Converter
output JP1 (DA0,GND,DA1,GND)



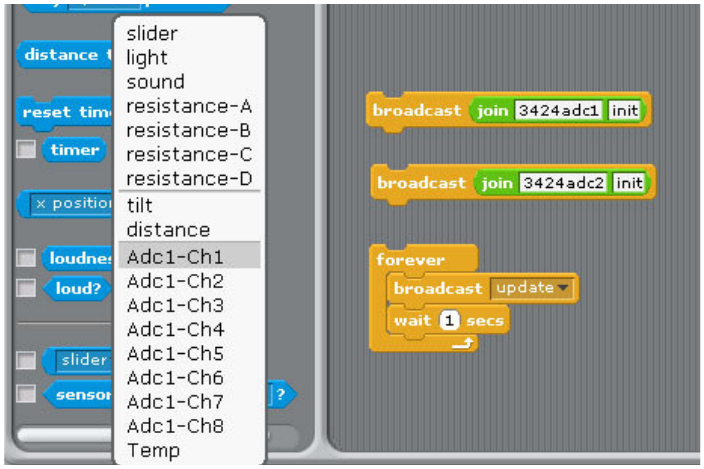
command "4802init" initial spi mcp4802
command "4802out" + (a or B) + output voltage
a -- DA0 , b -- DA1
(output voltage range 0.01 to 2)

21. Demo 16 i2c 18 bit MCP3424 ADC demo





MCP3424 U1 ~U4 --> 6c, 6d, 6e 6f
MCP23017 U13 U14 --> 21,22

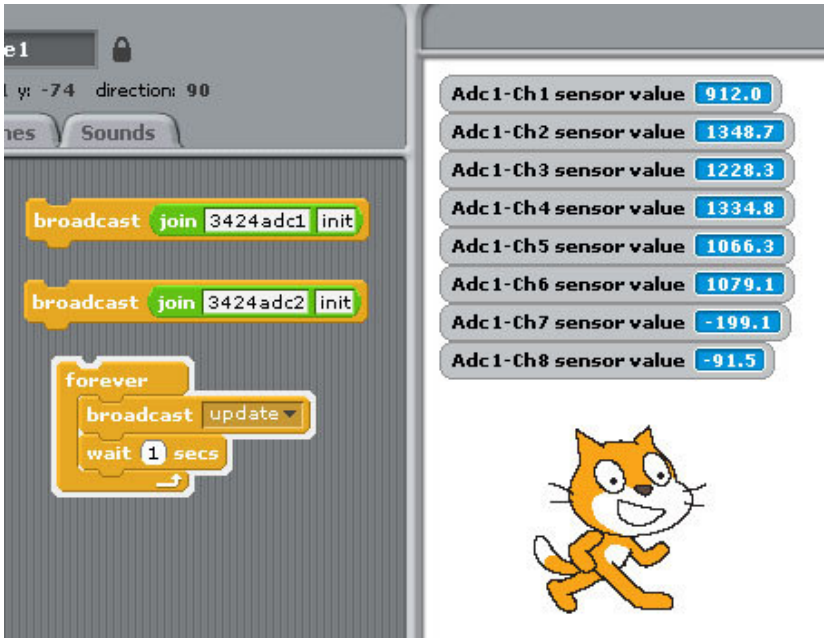


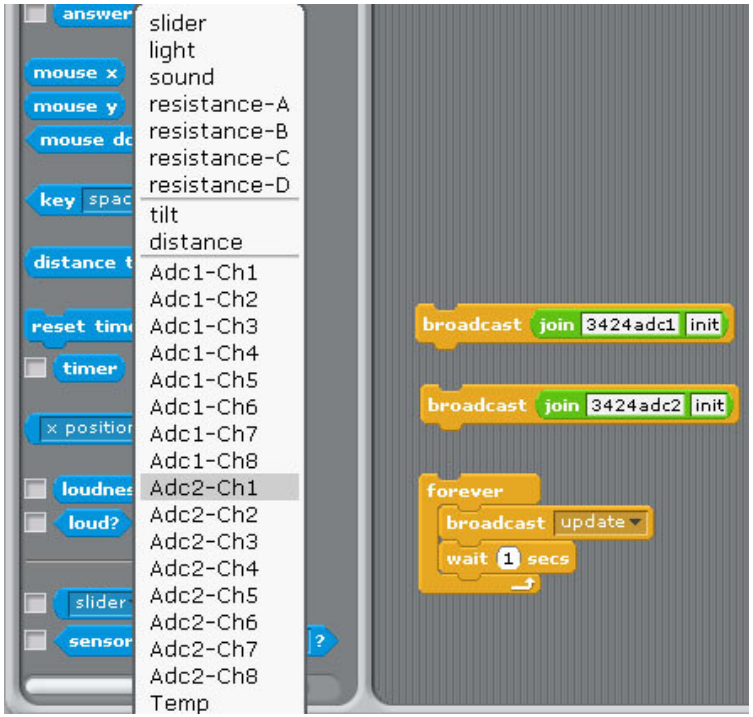
Adc1 address 6c, 6d adc2 address 6e, 6f

command "3424adc1init" initial i2c mcp3424

broadcast "Update" in Sensing ---> Slider ,

you will see "adc1-ch1" ~ "adc1-ch8" in the list





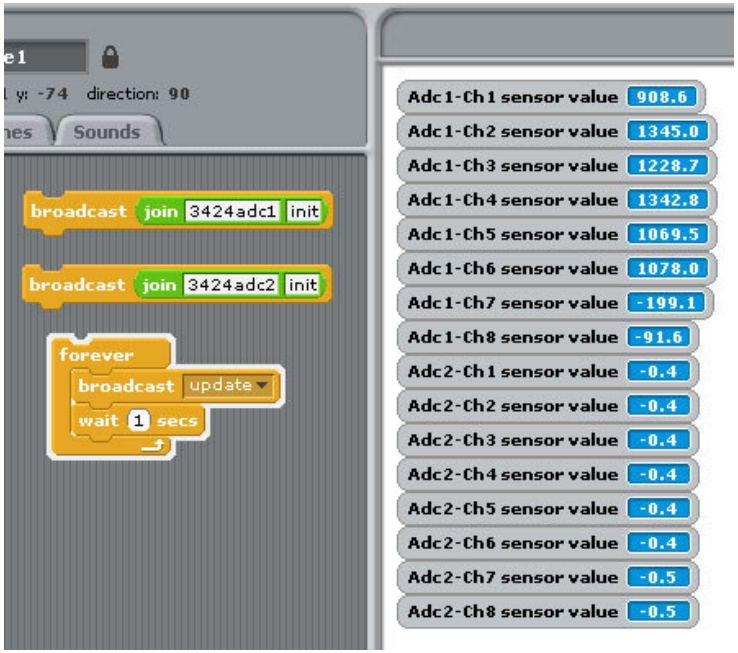
Adc1 address 6c, 6d adc2 address 6e, 6f

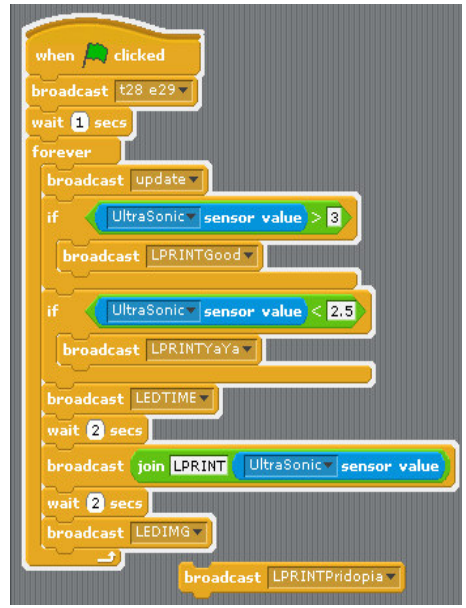
command "3424adc1init" initial i2c mcp3424

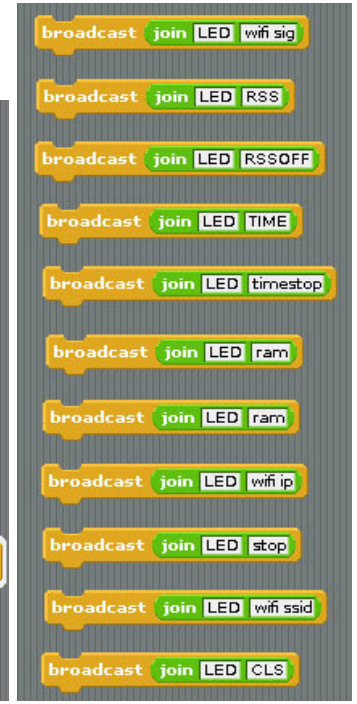
broadcast "Update" in Sensing ---> Slider ,

you will see "adc1-ch1" ~ "adc1-ch8" in the list

and "adc2-ch1" ~ "adc2-ch8" in the list







(1) command LED+ "TIME" display Time & Date
(2) command LED+ "IMG" display 24X16 image & Time/Date
(3) command LED + "message" display message
LEDthis is led matrix demo --> display "this is led matrix demo"
(4) command "LEDCLS" clean 24x 16 screen
(5) command LPRINT + "message"
Maximum display message 5X7 8 character in 24x16 Matrix

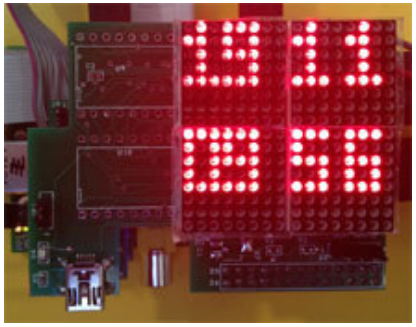
NEW COMMAND 14,Mar 2014 update

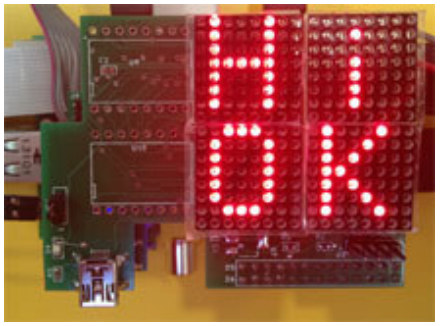
(6) command LED +"ram" Display RAM usage
(7) command LED +"stop" stop all display function
(8) command LED +"wifi ip" display WiFi IP
(9) command LED +"wifi ssid" display WiFi SSID
(10) command LED +"wifi sig" Display Wifi signal
(11) command LED +"RSS" RSS function on
(12) command LED +"RSSOFF" RSS function Off

22. Demo17 i2c 24X16 LED Matrix & 16x16 LED Matrix

(1)i2c 24x16 LED matrix

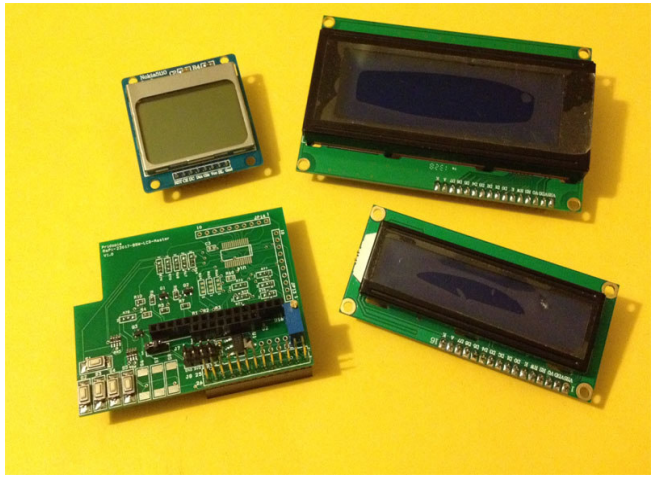
(2)i2c 16x16 LED matrix





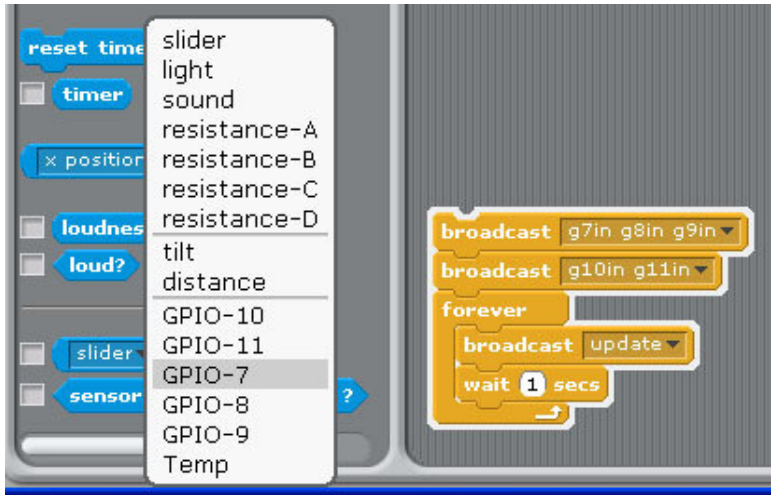
(1) command SLED+ "time" display Time & Date (3x5 font)
(2) command SLED+ "img" display 16X16 image
(3) command SLED + "message" display message
SLEDthis is led matrix demo --> display "this is led matrix demo"
(4) command "SLEDCLS" clean 16x16 screen
(5) command SPRINT + "message"
Maximum display message 5X7 4 character in 16x16 Matrix
NEW COMMAND 17,Mar 2014 update
(6) command LED +"ram" Display RAM usage
(7) command LED +"stop" stop all display function
(8) command LED +"wifi ip" display WiFi IP
(9) command LED +"wifi ssid" display WiFi SSID
(10) command LED +"wifi sig" Display Wifi signal
(11) command LED +"RSS" RSS function on
(12) command LED +"RSSOFF" RSS function Off

23. Demo18 GPIO 84x48 pixels monochrome LCD , GPIO 16x2 character LCD, GPIO 20x4 character LCD



our Rs-Pi LCD Master can plug-in 84x48 LCD , 16x2 LCD, 20x4 LCD

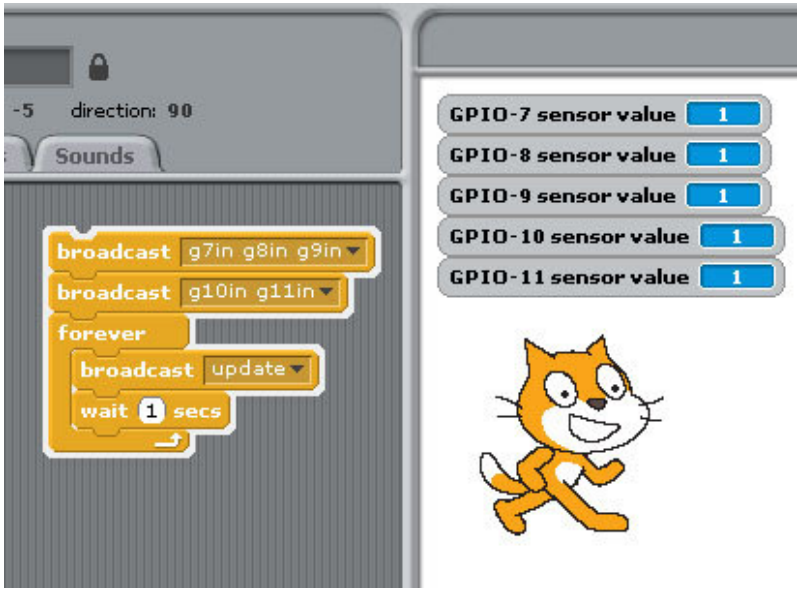
(1) 5 GPIO Switch GPIO 7,8,9,10,11



1.Setting GPIO 7,8,9,10,11 as input

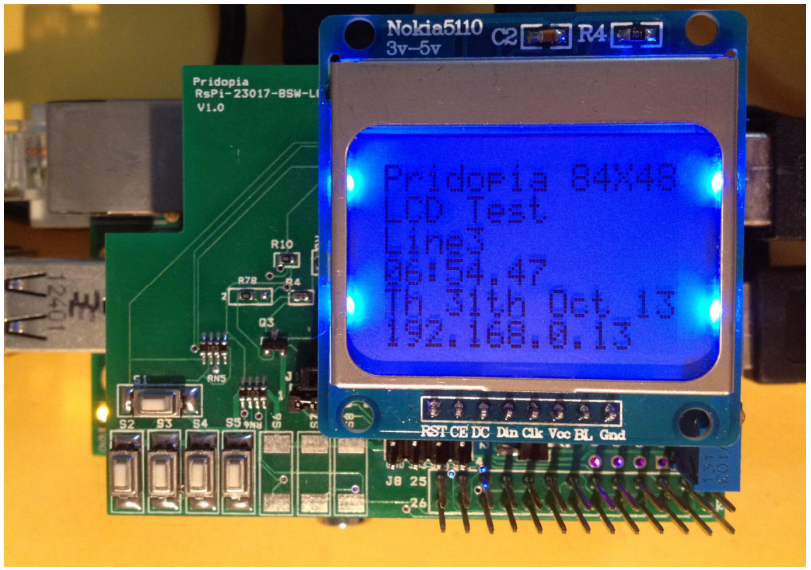
2. broadcast "Update"

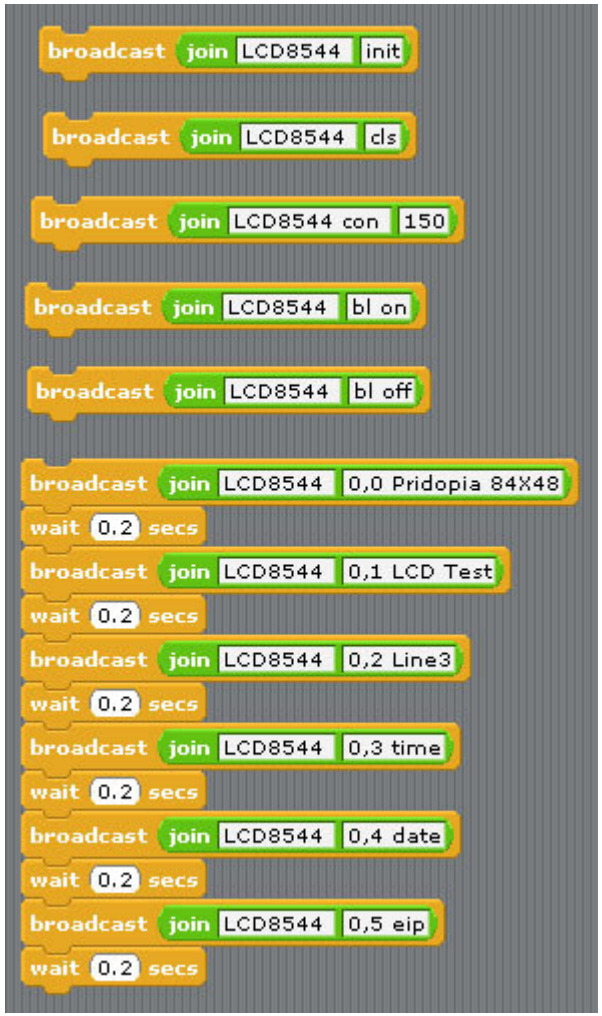
3. in Sensing --> Slider , you will see the GPIO-7 ,8, 9, 10, 11 in the list



(2) 84x48 pixels monochrome LCD

(2) 84x48 pixels LCD





command "LCD8544 " + "init" initial 84x48 LCD

command "LCD8544 " + "con " + " (0 -255)" LCD contrast

command "LCD8544 " + "bl " + "on/off" LCD back light ON / OFF

command "LCD8544 " + "cls" clean screen

command "LCD8544 " + "X , Y " + " MESSAGE" Display message to LCD in location x,y

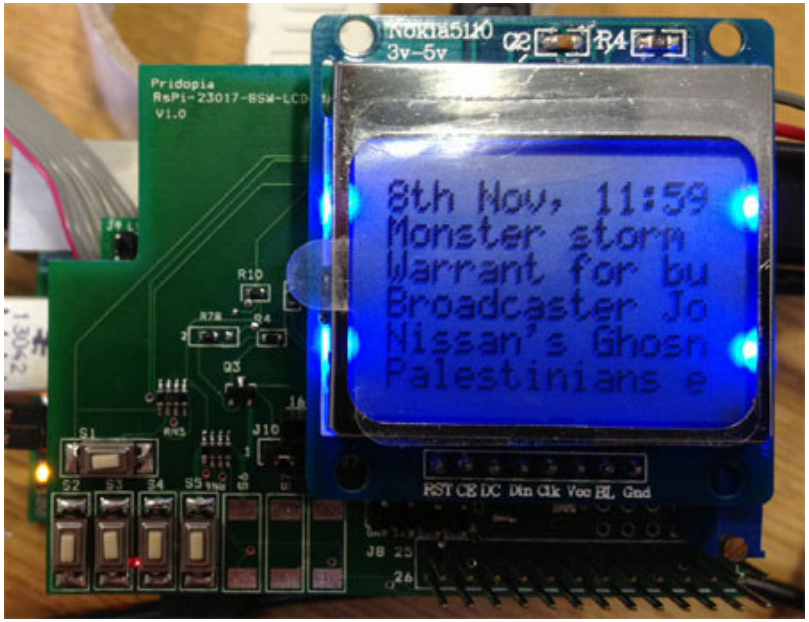
command "LCD8544 " + "X , Y " + " wip" Display WiFi IP to LCD in location x,y

command "LCD8544 " + "X , Y " + " eip" Display Ethernet IP to LCD in location x,y

command "LCD8544" + "X , Y " + " time" Display Time to LCD in location x,y

command "LCD8544 " + "X , Y " + " date" Display Time to LCD in location x,y

LCD 84x48 Top 5 news from BBC RSS



command "LCD8544 init" initial GPIO 84x48 LCD

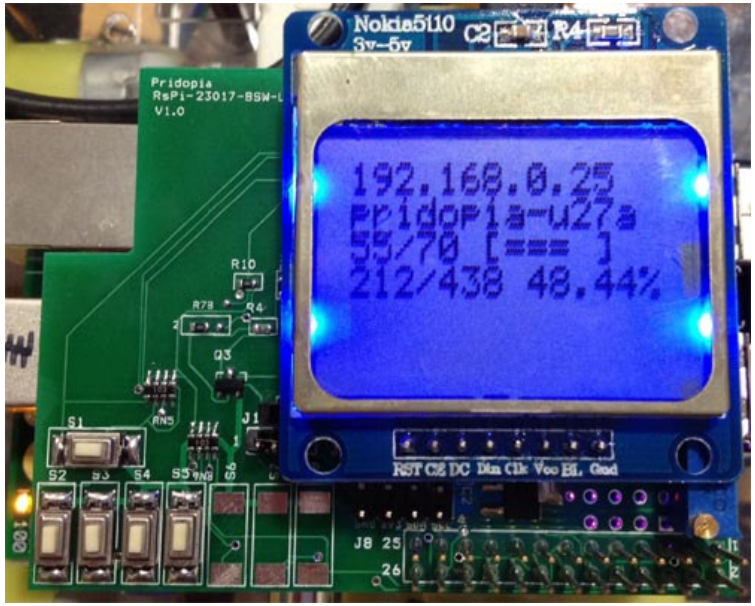
command "LCD8544 con 170" (0 -255)" LCD contrast

command "LCD8544 "rss"+ " speed" Active RSS function (need have internet)

LCD8544 rss 5 speed (1 to 20 - 1 is fast , 20 is slow)

command "LCD8544 rss off" stop RSS function

84x48 New command ** 8,Jan, 2014 update



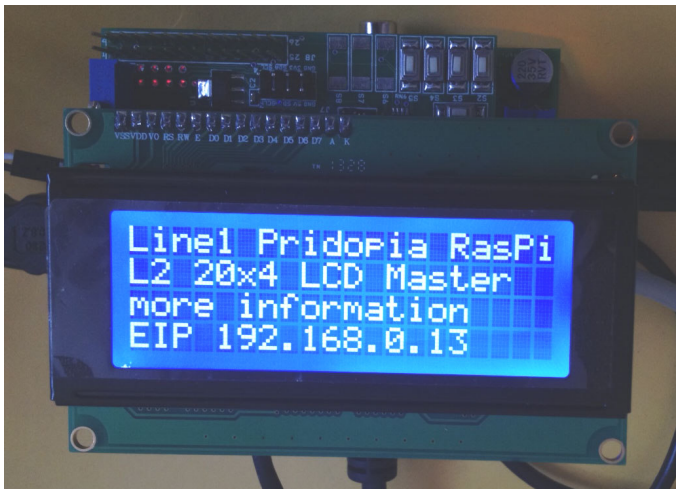
command "LCD8544 " + "X , Y " + " ram" Display RAM usage in location x,y

command "LCD8544 " + "X , Y " + " wifi" + " ip" Display WiFi IP in location x,y

command "LCD8544 " + "X , Y " + " wifi" + " ssid" Display WiFi SSID in location x,y

command "LCD8544 " + "X , Y " + " wifi" + " gsig" Display wifi signal with bar chart in location x,y

(3) 20x4 Character LCD



(1) command

command "LCD2004g " + "init" initial GPIO 20x4 LCD

command "LCD2004 " + "X , Y " + " MESSAGE" Display message to LCD in location x,y

command "LCD2004 " + "cls" clean screen

command "LCD2004 " + "bl " + "on/off" LCD back light ON / OFF

command "LCD2004 " + "X , Y " + " wip" Display WiFi IP to LCD in location x,y

command "LCD2004 " + "X , Y " + " eip" Display Ethernet IP to LCD in location x,y

command "LCD2004 " + "X , Y " + " time" Display Time to LCD in location x,y

command "LCD2004" + "X , Y " + " date" Display Time to LCD in location x,y

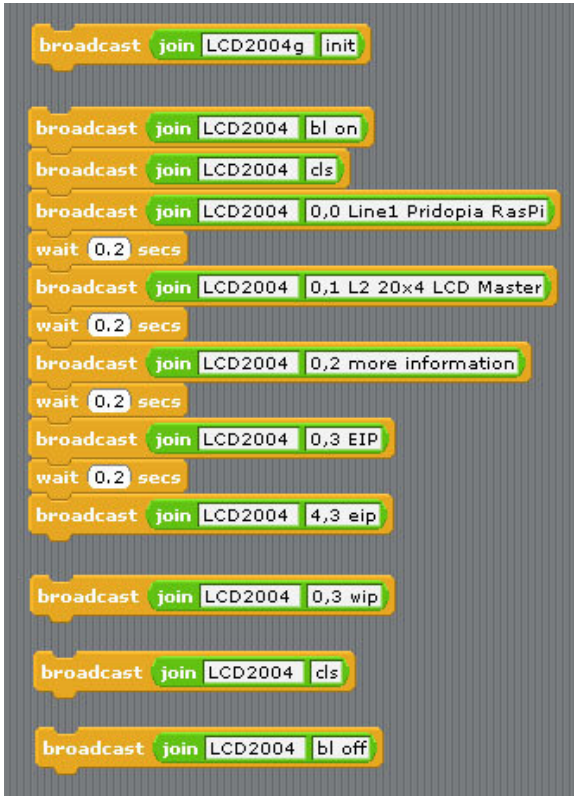
(2) RSS function command "LCD2004g init" initial GPIO 20x4 LCD

command "LCD2004 bl on" Turn On LCD back light

command "LCD2004 "rss"+ " speed" Active RSS function (need have internet)

LCD2004 rss 5 speed (1 to 20 - 1 is fast , 20 is slow)

command "LCD2004 rss off" stop RSS function



(4) 20x4 Character LCD

RSS function (Scrolling text Display top 3 News message)

BBC world news RSS feed

Date & Time in first Line, 2,3,4 line show top 3 BBC RSS news

(4) GPIO 20x4 new command ** 8,Jan, 2014 update

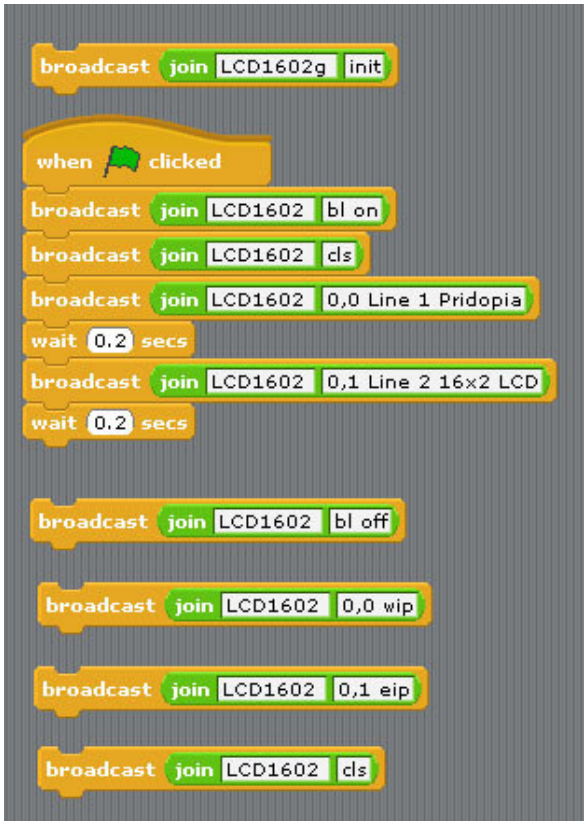
command "LCD2004 " + "X , Y " + " ram" Display RAM usage in location x,y

command "LCD2004 " + "X , Y " + " wifi" + " ip" Display WiFi IP in location x,y

command "LCD2004 " + "X , Y " + " wifi" + " ssid" Display WiFi SSID in location x,y

command "LCD2004 " + "X , Y " + " wifi" + " gsig" Display wifi signal with bar chart in location x,y

(5) 16x2 Character LCD



command "LCD1602g " + "init" initial GPIO 16x2 LCD

command "LCD1602 " + "X , Y " + " MESSAGE" Display message to LCD in location x,y

command "LCD1602 " + "cls" clean screen

command "LCD1602 " + "bl " + "on/off" LCD back light ON / OFF

command "LCD1602 " + "X , Y " + " wip" Display WiFi IP to LCD in location x,y

command "LCD1602 " + "X , Y " + " eip" Display Ethernet IP to LCD in location x,y

command "LCD1602 " + "X , Y " + " time" Display Time to LCD in location x,y

command "LCD1602 " + "X , Y " + " date" Display Time to LCD in location x,y

16x2 RSS function

RSS function (Scrolling text Display top 4 News message)

message format

Line1 --> Time + message 1 + " ->-> " + Message 3 + Time

Line2 --> Time + message 2 + " ->-> " + Message 4 + Time

BBC world news RSS feed

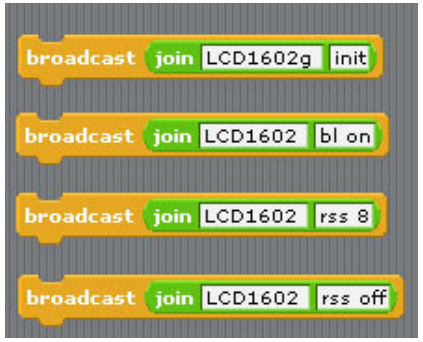
command "LCD1602g init" initial GPIO 16x2 LCD

command "LCD1602 bl on" turn on LCD back light

command "LCD1602 "rss"+ " speed" Active RSS function (need have internet)

LCD1602 rss 8 speed (1 to 20 - 1 is fast , 20 is slow)

command "LCD1602 rss off" stop RSS function



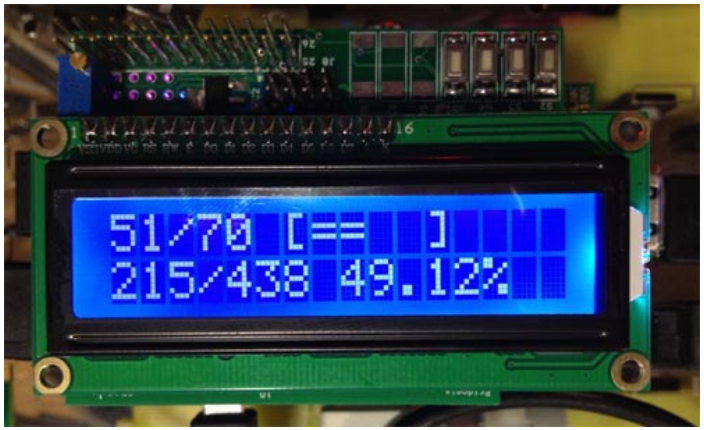
(6) **GPIO 16x2 LCD new command ** 08,Jan,2014 update**

command "LCD1602 " + "X , Y " + " ram" Display RAM usage in location x,y

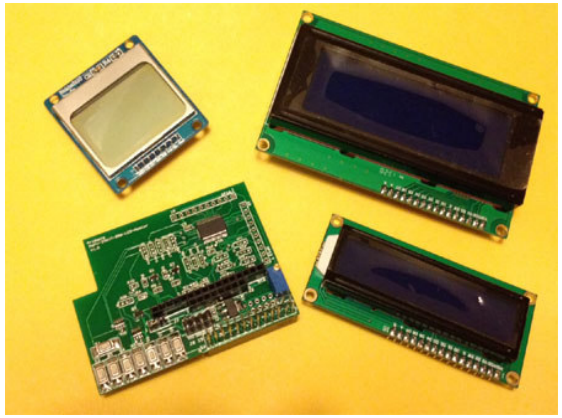
command "LCD1602 " + "X , Y " + " wifi" + " ip" Display WiFi IP in location x,y

command "LCD1602 " + "X , Y " + " wifi" + " ssid" Display WiFi SSID in location x,y

command "LCD1602 " + "X , Y " + " wifi" + " gsig" Display wifi signal with bar chart in location x,y

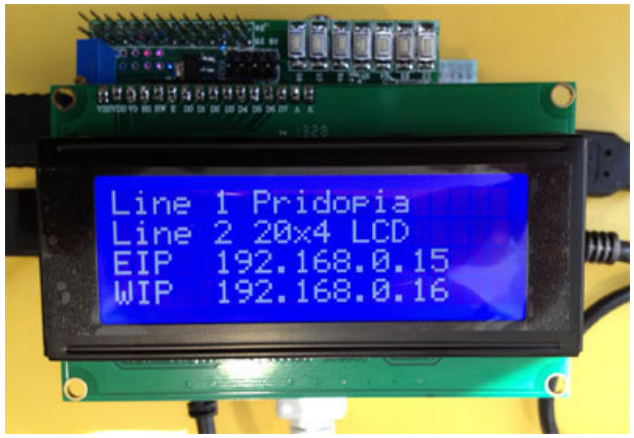


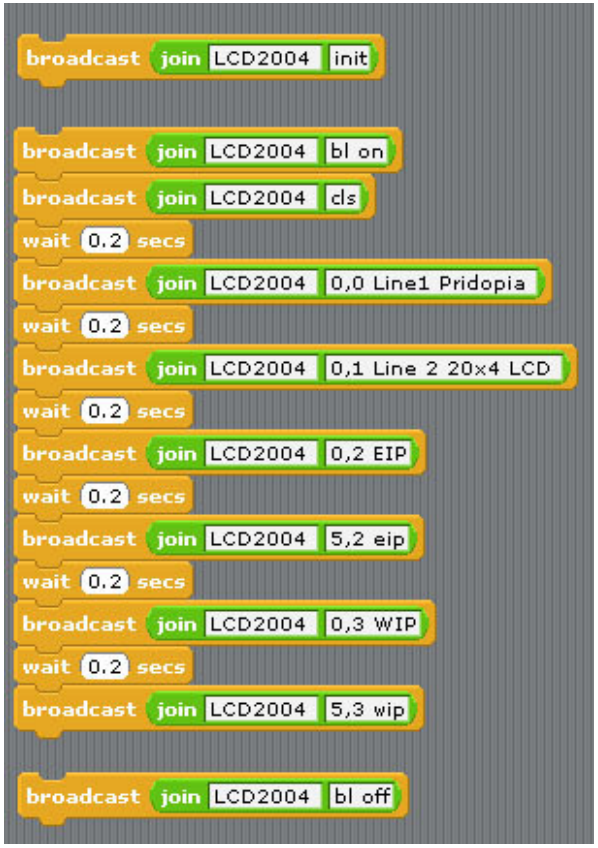
24. Demo19 GPIO 84x48 LCD , i2c 16x2 LCD, i2c 20x4 LCD



our Rs-Pi i2c LCD Master can plug-in 84x48 LCD , 16x2 LCD, 20x4 LCD

(1) i2c 20x4 character LCD





command "LCD2004 " + "init" initial i2c 20x4 LCD

command "LCD2004 " + "X , Y " + " MESSAGE" Display message to LCD in location x,y

command "LCD2004 " + "cls" clean screen

command "LCD2004 " + "bl " + "on/off" LCD back light ON / OFF

command "LCD2004 " + "X , Y " + " wip" Display WiFi IP to LCD in location x,y

command "LCD2004 " + "X , Y " + " eip" Display Ethernet IP to LCD in location x,y

command "LCD2004 " + "X , Y " + " time" Display Time to LCD in location x,y

command "LCD2004" + "X , Y " + " date" Display Time to LCD in location x,y

(2) 20x4 RSS function

RSS function (Scrolling text Display top 3 News message)

BBC world news RSS feed

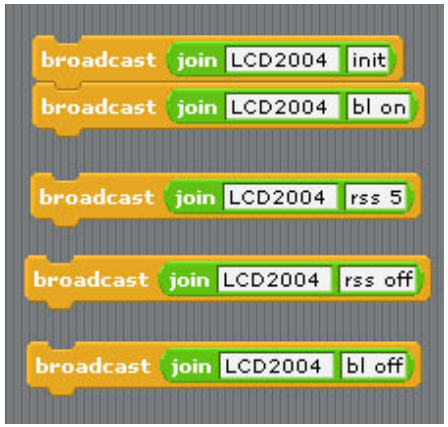
command "LCD2004 init" initial i2c 20x4 LCD

command "LCD2004 bl on" turn on LCD back light

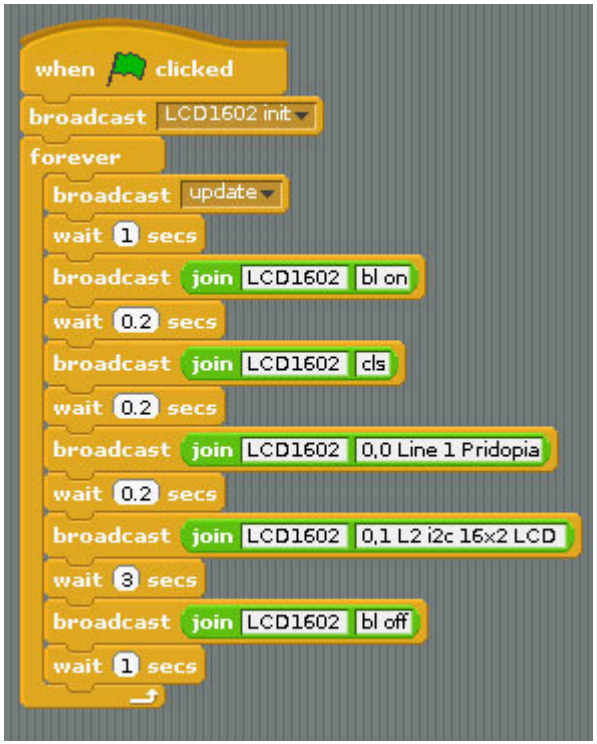
command "LCD2004 "rss"+ " speed" Active RSS function (need have internet)

LCD2004 rss 5 speed (1 to 20 - 1 is fast , 20 is slow)

command "LCD2004 rss off" stop RSS function



(2) i2c 16x2 character LCD



command "LCD1602 " + "init" initial i2c 16x2 LCD

command "LCD1602 " + "X , Y " + " MESSAGE" Display message to LCD in location x,y

command "LCD1602 " + "cls" clean screen

command "LCD1602 " + "bl " + "on/off" LCD back light ON / OFF

command "LCD1602 " + "X , Y " + " wip" Display WiFi IP to LCD in location x,y

command "LCD1602 " + "X , Y " + " eip" Display Ethernet IP to LCD in location x,y

command "LCD1602 " + "X , Y " + " time" Display Time to LCD in location x,y

command "LCD1602 " + "X , Y " + " date" Display Time to LCD in location x,y

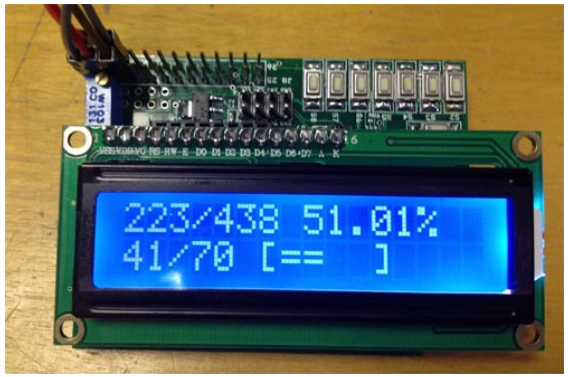
**** new command 08,Jan,2014 update for address 21 only**

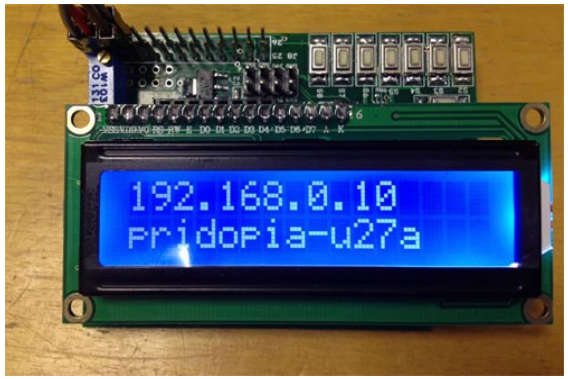
command "LCD1602 " + "X , Y " + " ram" Display RAM usage in location x,y

command "LCD1602 " + "X , Y " + " wifi" + " ip" Display WiFi IP in location x,y

command "LCD1602 " + "X , Y " + " wifi" + " ssid" Display WiFi SSID in location x,y

command "LCD1602 " + "X , Y " + " wifi" + " gsig" Display wifi signal with bar chart in location x,y





16x2 LCD RSS function

RSS function (Scrolling text Display top 4 News message)

message format

Line1 --> Time + message 1 + " ->-> " + Message 3 + Time

Line2 --> Time + message 2 + " ->-> " + Message 4 + Time

BBC world news RSS feed

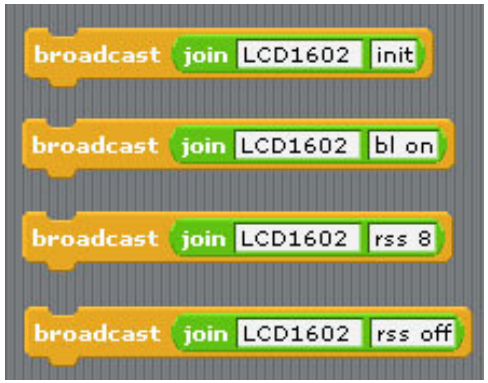
command "LCD1602 init" initial i2c 16x2 LCD

command "LCD1602 bl on" turn on LCD back light

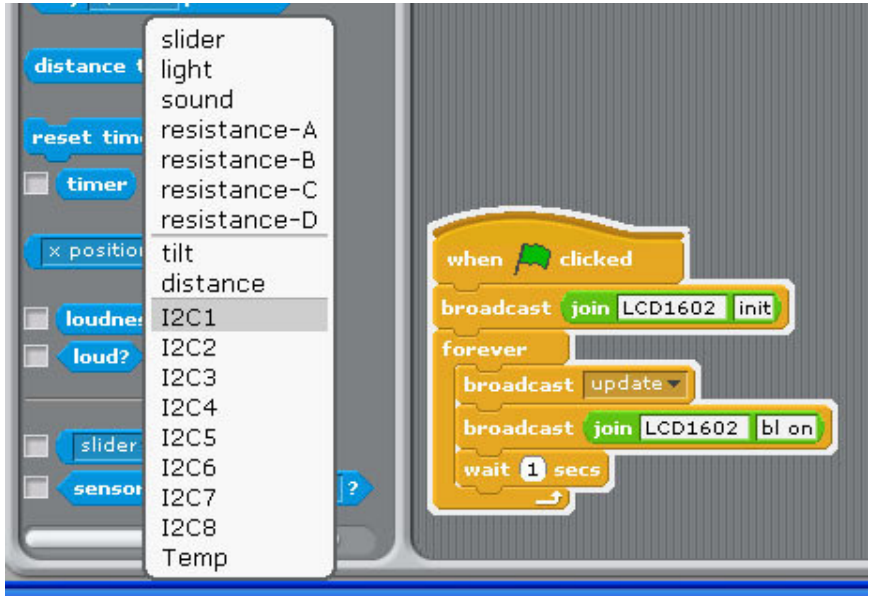
command "LCD1602 "rss"+ " speed" Active RSS function (need have internet)

LCD1602 rss 8 speed (1 to 20 - 1 is fast , 20 is slow)

command "LCD1602 rss off" stop RSS function



(3) i2c LCD1602 & LCD2004 8 switch detect in scratch



(1) broadcast "LCD1602 init" for 16x2 LCD

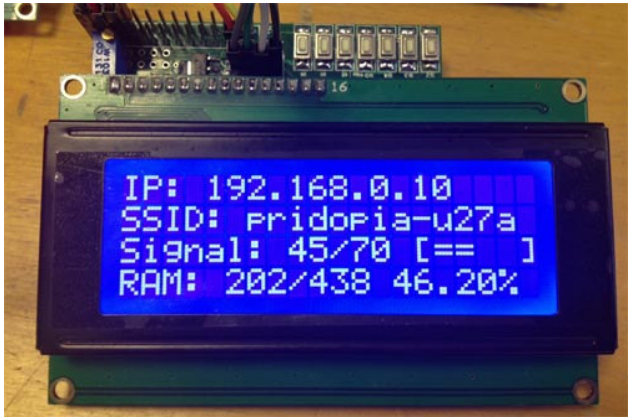
broadcast "LCD2004 init" for 20x4 LCD

(2)after broadcast "Update"

in Sensing --> Slider , you will see the

" I2C1, I2C2, I2C3, I2C4, I2C5, I2C6, I2C7, I2C8 " in the list

(4)i2c & GPIO 20x4 LCD new command



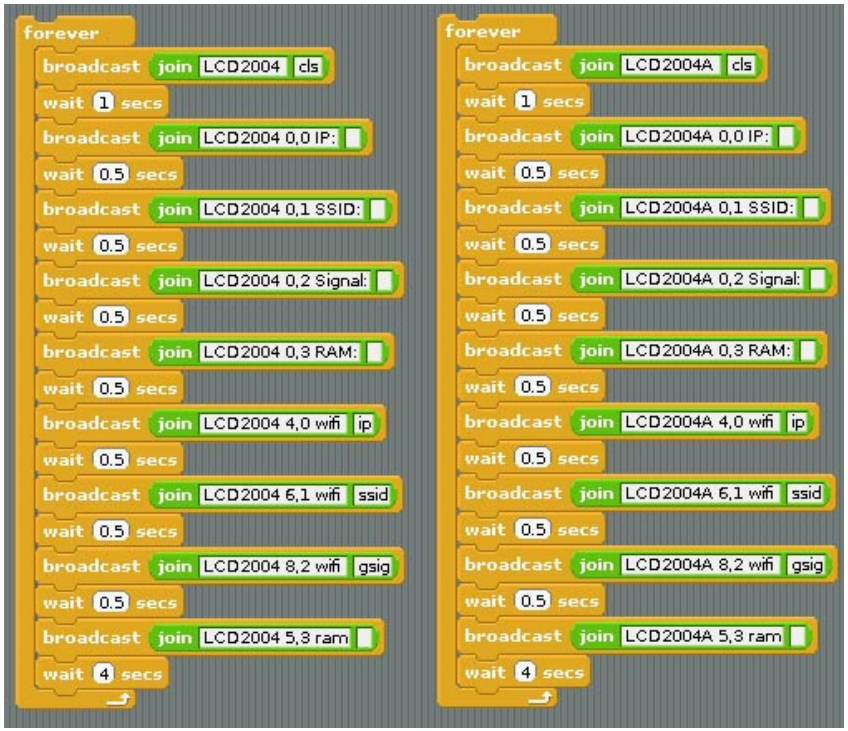
command "LCD2004 " + "X , Y " + " ram" Display RAM usage in location x,y

command "LCD2004 " + "X , Y " + " wifi" + " ip" Display WiFi IP in location x,y

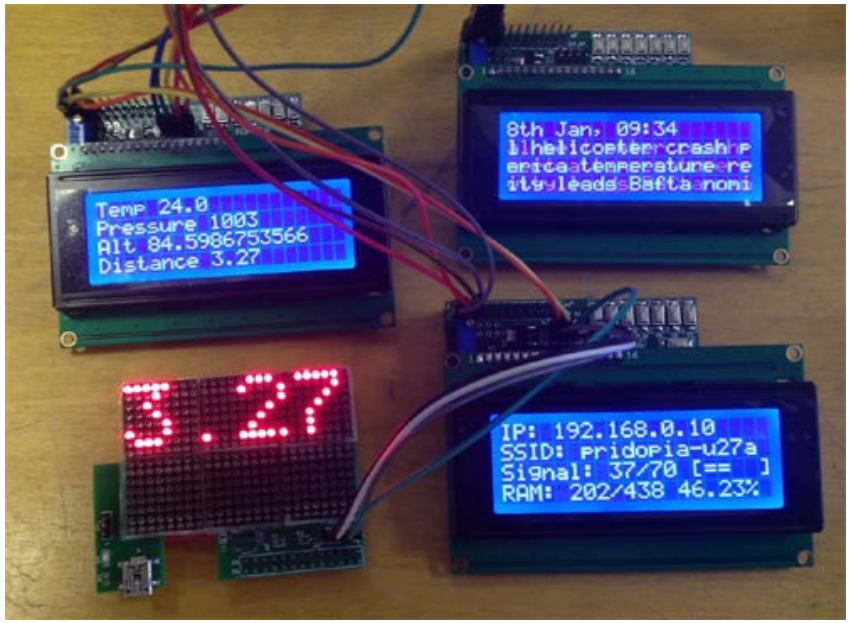
command "LCD2004 " + "X , Y " + " wifi" +" ssid" Display WiFi SSID in location x,y

command "LCD2004 " + "X , Y " + " wifi" + " gsig" Display wifi signal with bar chart in location x,y

also work with i2c LCD module "LCD2004A" ,"LCD2004B", "LCD2004C" ,"LCD2004D"



(5) multi i2c 20x4 LCD demo



A = ADDRESS 24, B = ADDRESS 25, C = ADDRESS 26, D = ADDRESS 27

command "LCD2004 " + "init" initial i2c 20x4 LCD address 21

command "LCD2004A " + "init" initial i2c 20x4 LCD address 24

command "LCD2004B " + "init" initial i2c 20x4 LCD address 25

command "LCD2004C " + "init" initial i2c 20x4 LCD address 26

command "LCD2004D " + "init" initial i2c 20x4 LCD address 27

command "LCD2004" "Address" + "X , Y " + " MESSAGE" Display message in location x,y

command "LCD2004" + "address" +" cls" clean screen

command "LCD2004" +"address" + "bl " + "on/off" LCD back light ON / OFF

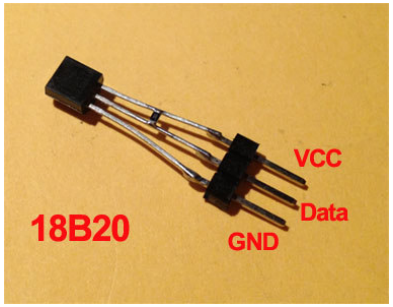
command "LCD2004" + "address" +"X , Y " + " wip" Display WiFi IP to LCD in location x,y

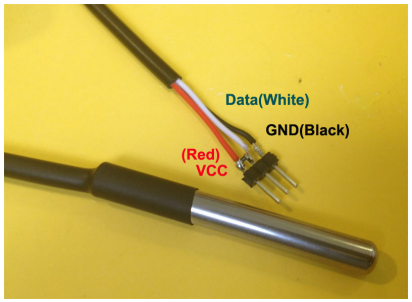
command "LCD2004" +"address" + "X , Y " + " eip" Display Ethernet IP to LCD in location x,y

command "LCD2004" +"address" + "X , Y " + " time" Display Time to LCD in location x,y

command "LCD2004" + "address" +"X , Y " + " date" Display Time to LCD in location x,y

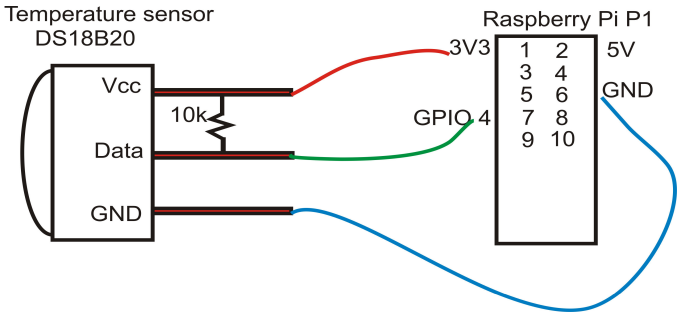
25. Demo 20 (1) 1-Wire 18B20 temp Sensor

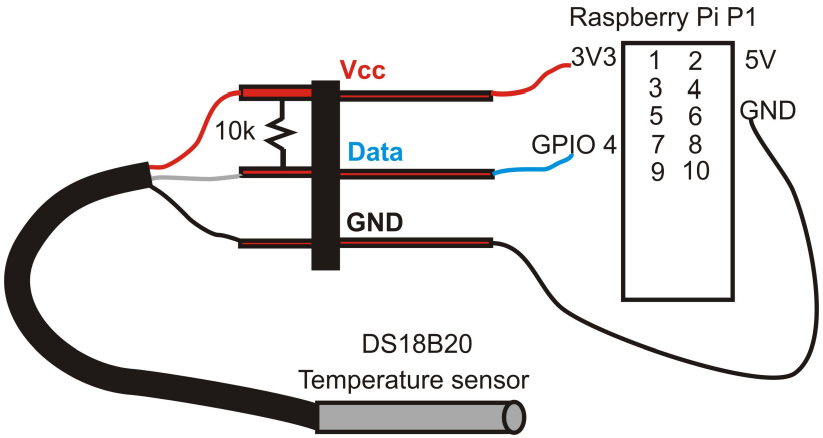


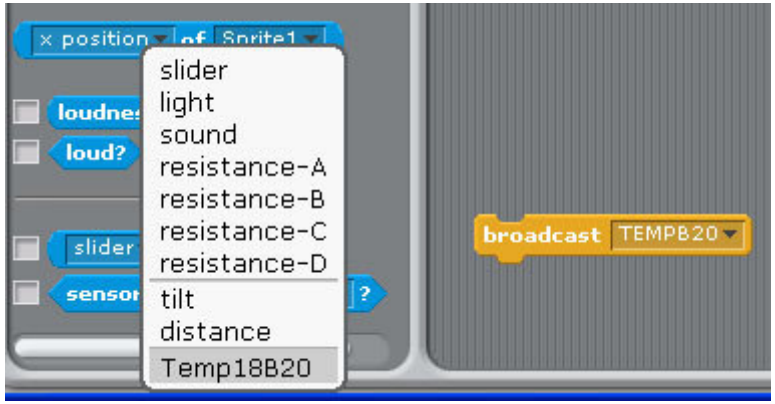


VCC to 3.3V P1 pin1, Data to P1 pin7(GPIO4) , GND to P1 pin6(GND)

Data & VCC we have one 10K SMD Resister





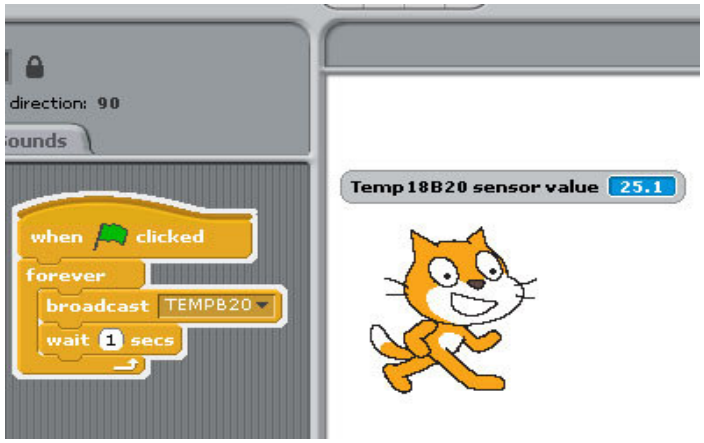


Command " TEMPB20" to active 1-wire in pin7(GPIO4)

1.broadcast ""TEMPB20"

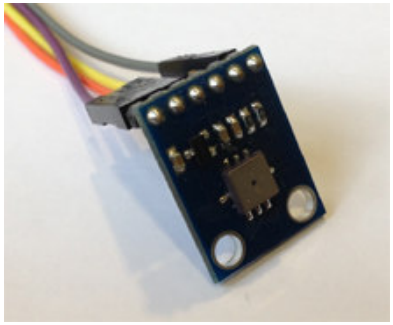
2. after broadcast in Sensing --> Slider ,

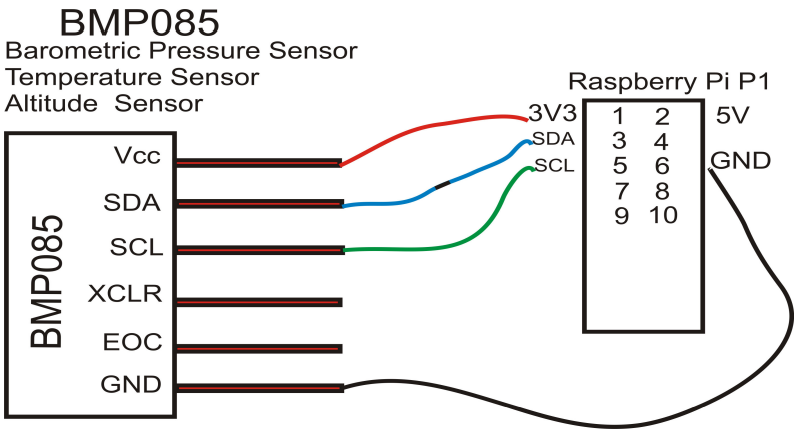
you will see the " Temp18B20" in the list



(2) i2c address 77 BMP085 sensor

BMP085 Barometric Pressure/Temperature/Altitude Sensor

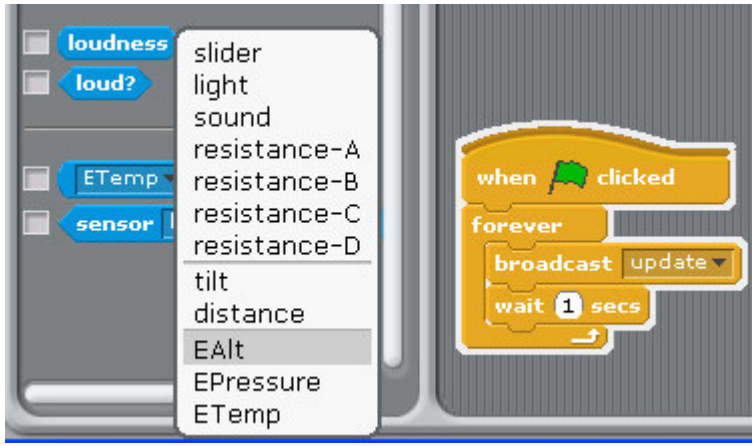




BMP085 is a kind of pressure sensor with high precision, low en ergy, which
can be used in mobile devices. Its excellent performance makes t he absolute precision reach the minimum0.03hPa, and it is with v ery low power consumption, only 3u A. sing powerful 8-
pin ceramic leadless chip carrier (LCC) thin package, BMP085 can be connected directly with a variety of microprocessor s through I2C bus .

Main features:

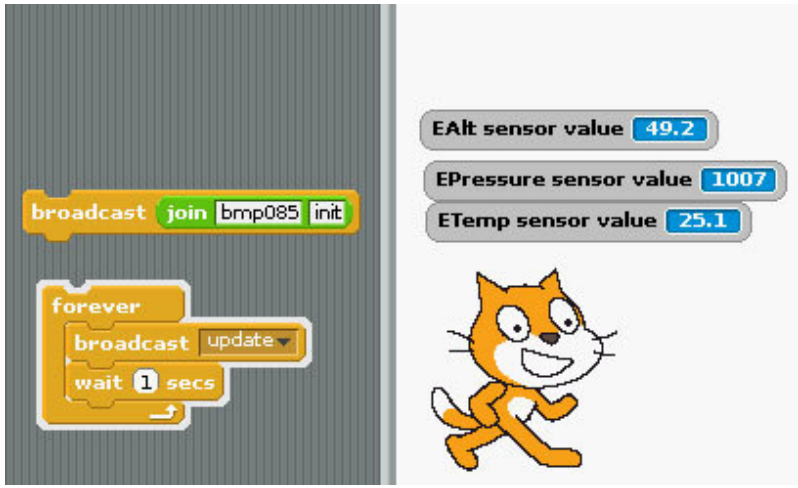
- Pressure range: 300 — 1100hPa
  (at an altitude of 9000 meters to 500 meters)
- Supply voltage: 1.8V — 3.6V （ VDDA ） 1.62V — 3.6V （ VDDD ）
- LCC8 package:ceramic leadless chip carrier(LCC)
- Size: 5.0mmx5.0x1.2mm
- Low consumption: 5 μ A in standard mode
- High accuracy:resolution ratio is 0.06hPa
  (0.5 meter) in low power consumption mode; 0.03hPa
  (0.25 meter) in high linear mode
- Temperature output



1.Command "bmp085init" initial i2c bmp085 address 77

2.after broadcast ""Update" in Sensing --> Slider ,

you will see the " EAlt" "Epressure" "ETemp" in the list

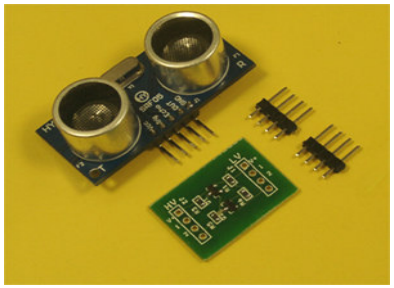


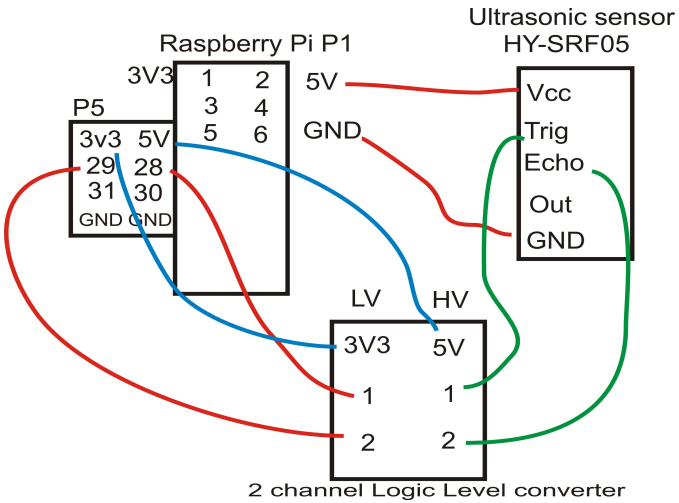
(3) Ultrasonic distance sensor (HY-SRF05) with 2channel Logic Level converter

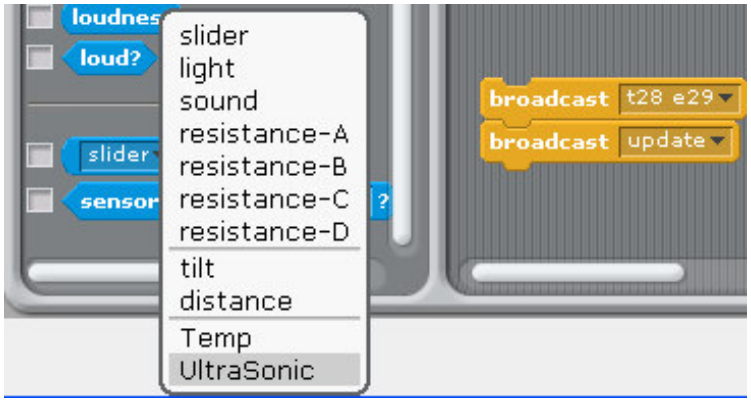
Use two GPIO Trig(T) Echo(E)

Update --> check distance and update

T28 --> Trig use GPIO28 E29 --> Echo use GPIO29



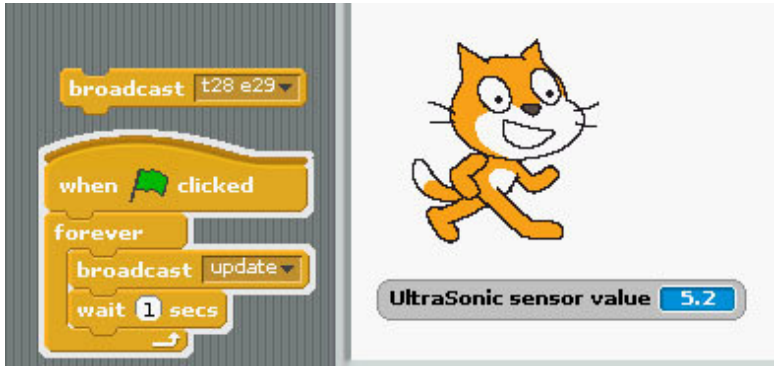


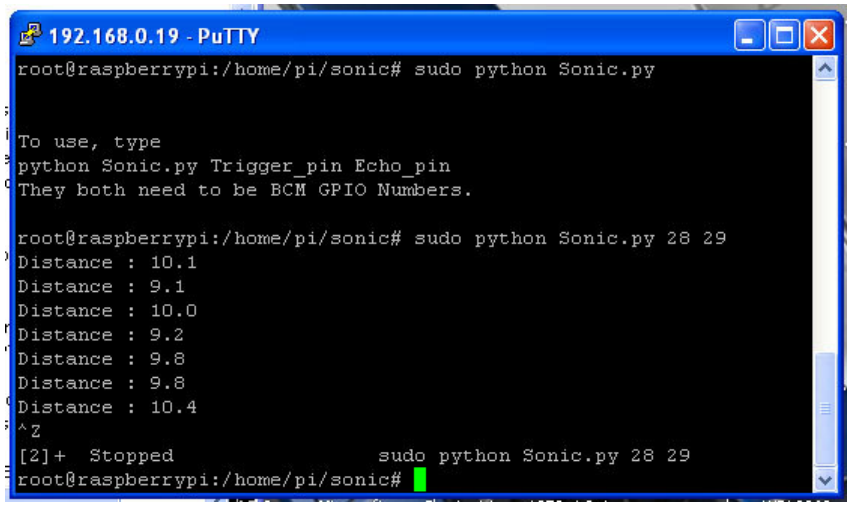


after broadcast "t28 e29 " & "Update"

T28 Trig in GPIO28 E29 Echo in GPIO29

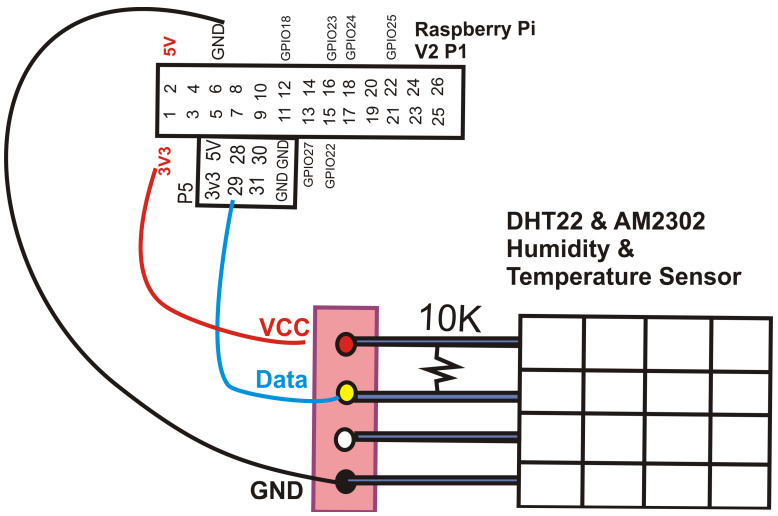
in Sensing --> Slider , you will see the " UltraSonic" in the list



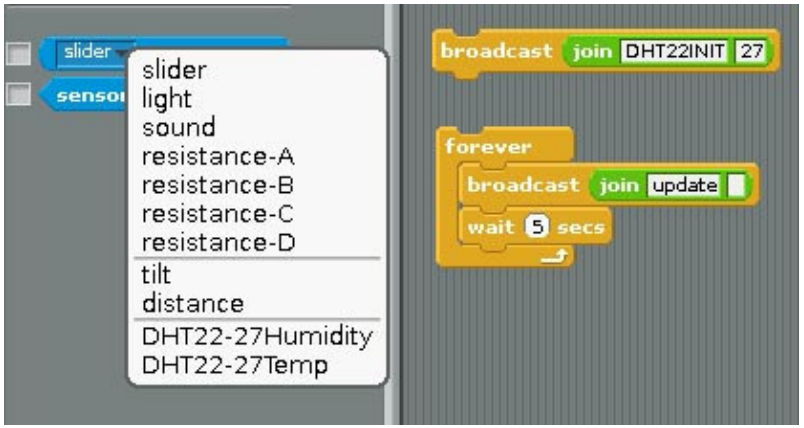


Python Sonic.py test program

(4) DHT22/AM2302 Digital Temperature & Humidity Sensor

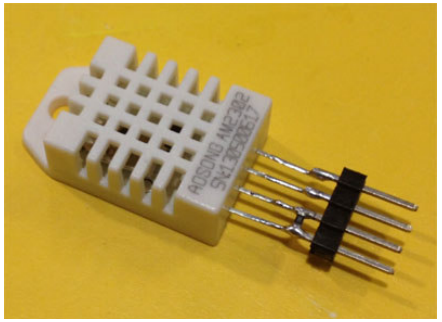


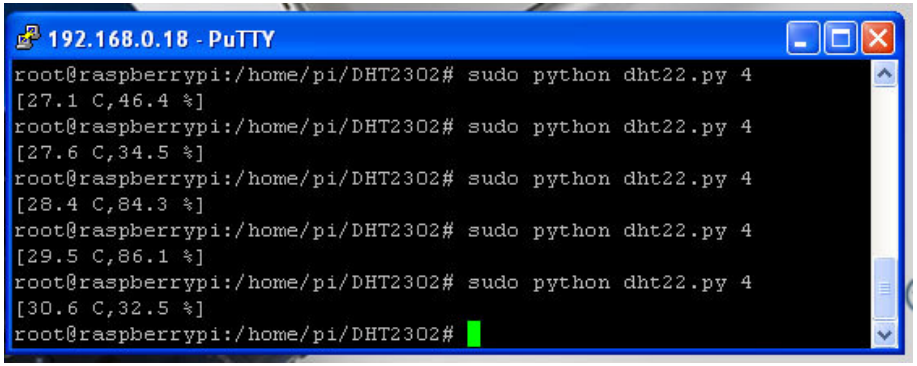
Type: AM2302 /DHT22
Voltage range: 3.5V-6V DC Humidity range: 0%-100%RH
Temperature range: -40°C to 80°C
Humidity accuracy: ±2RH% Temperature accuracy: ±0.5°C
Size: 3*1.3*0.6 cm



1.broadcast "DHT22INIT" +"27" GPIO number

in Sensing --> Slider , you will see the "DHT22-27Humidity" & "DHT22-27Temp" in the list



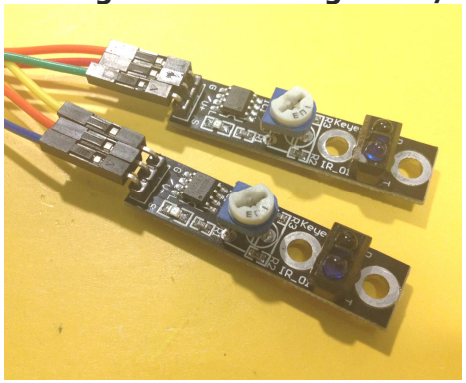


DHT22 python program

"sudo python dht22.py 4 " 4 is GPIO number (P1 pin7 GPIO 4) you can use other GPIO pin number

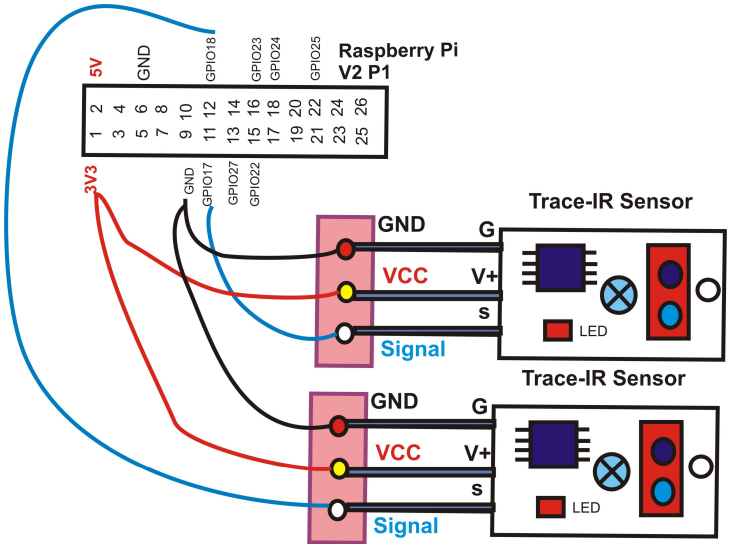
(5) Trace Line Hunting IR Sensor

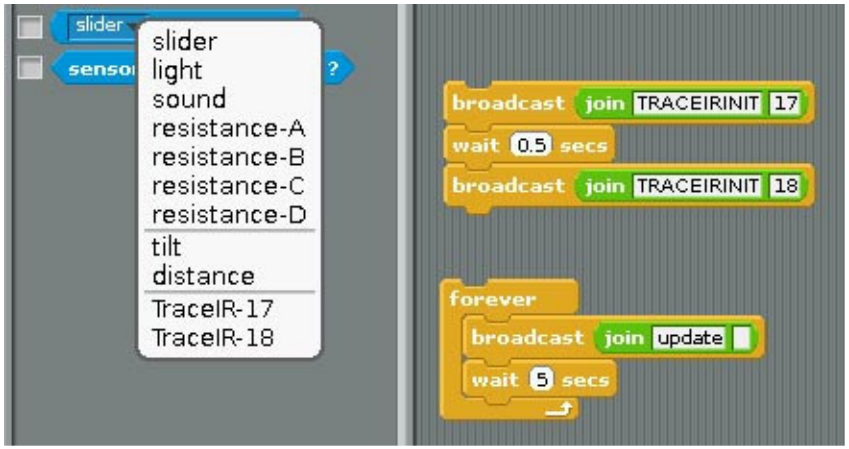
Intelligent Car Tracing Black/White Line Hunting Sensor



Description:

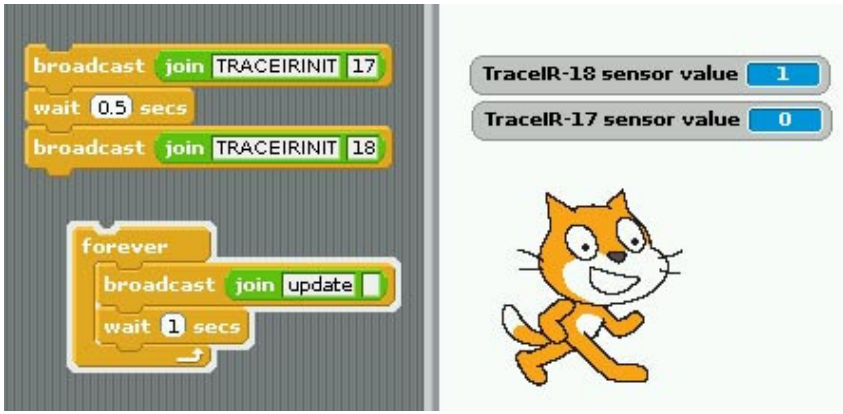
- VCC/OUT/GND pin connector
- Power Supply: 3.3V
- Working Current: 15-20mA at 3.3V
- Output electrical level signal: Low level when detecting objects/ high level when no objects/0 or 1 decides if objects exist)
- Dimensions: 1.81 x 0.39 x 0.35 inch / 4.6 x 1.0 x 0.9 cm



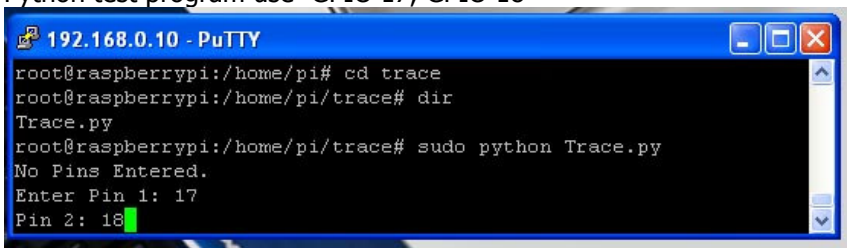


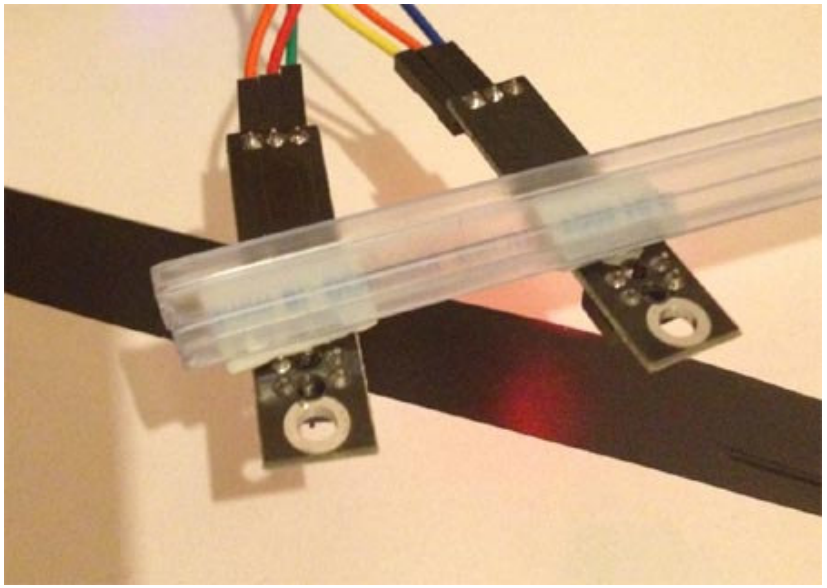
broadcast "TRACEIRINIT" +"17" GPIO number & "TRACEIRINIT" +"18"

in Sensing --> Slider , you will see the "TraceIR-17" & "TraceIR-18" in the list



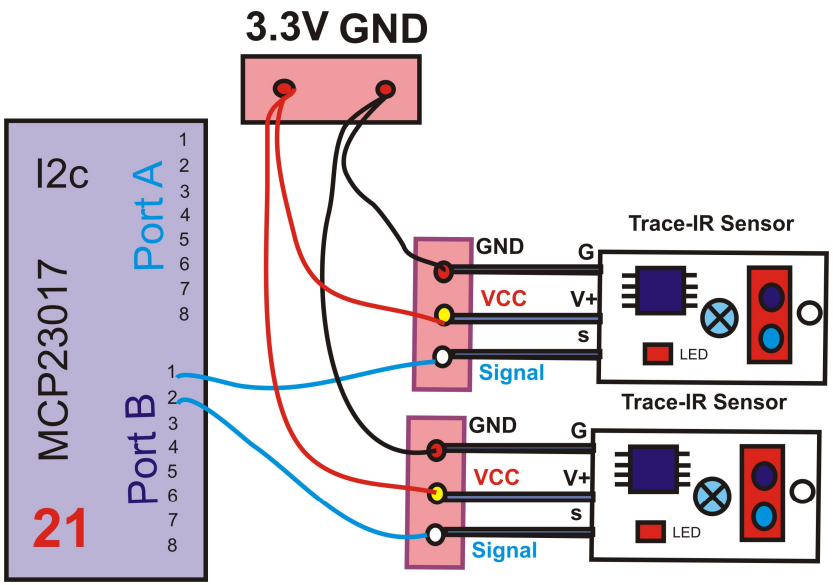
Python test program use GPIO 17, GPIO 18

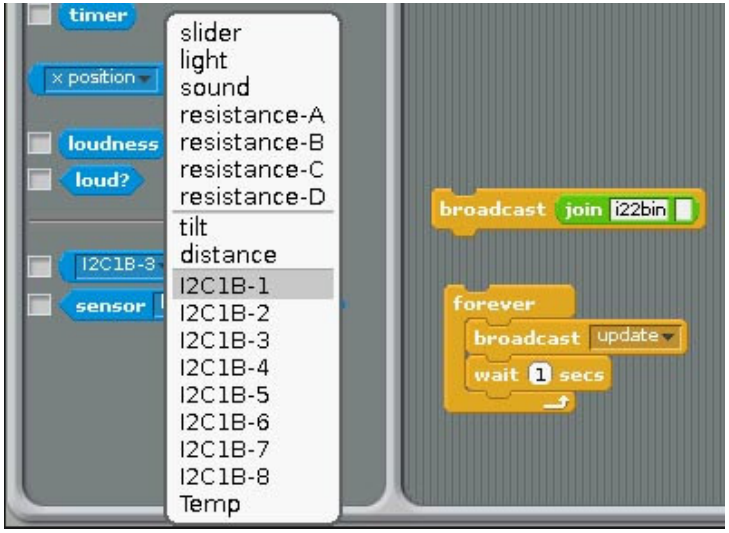




Detect the black line & white area

Intelligent Car Tracing Black/White Line Hunting Sensor
connect to 23017 i2c GPIO Port B





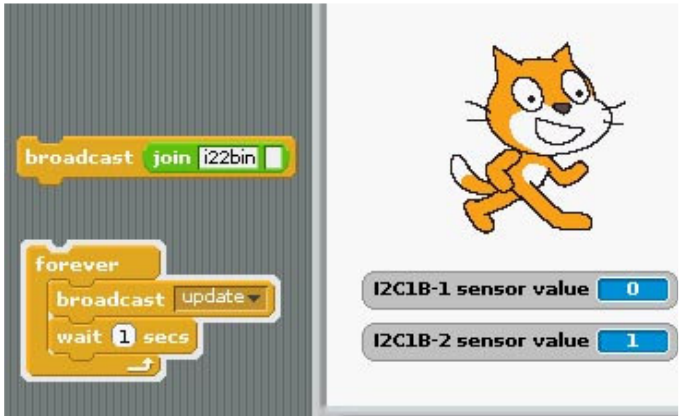

```
Command "i2"+ "address(1-8)" + "a" +"in" for Port A
Command "i2"+ "address(1-8)" + "b" +"in" for Port B
 Address 20 --> 1 21 --> 2 22-->3 23 -->4
 Address 24 --> 5 25 --> 6 26-->7 27 -->8
```

```
command "i22bin" initial address 21, Port B as input
```

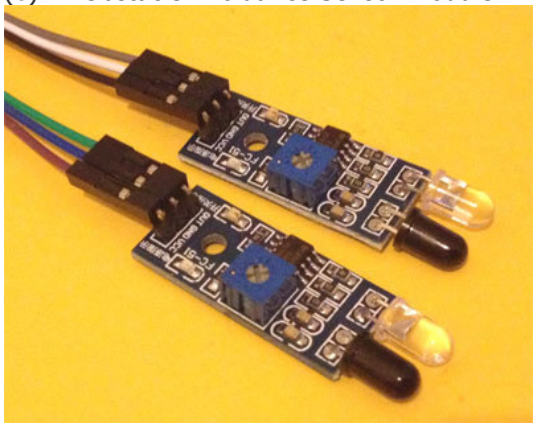
```
(1)"i22bin" initial address 21, Port B as input
```

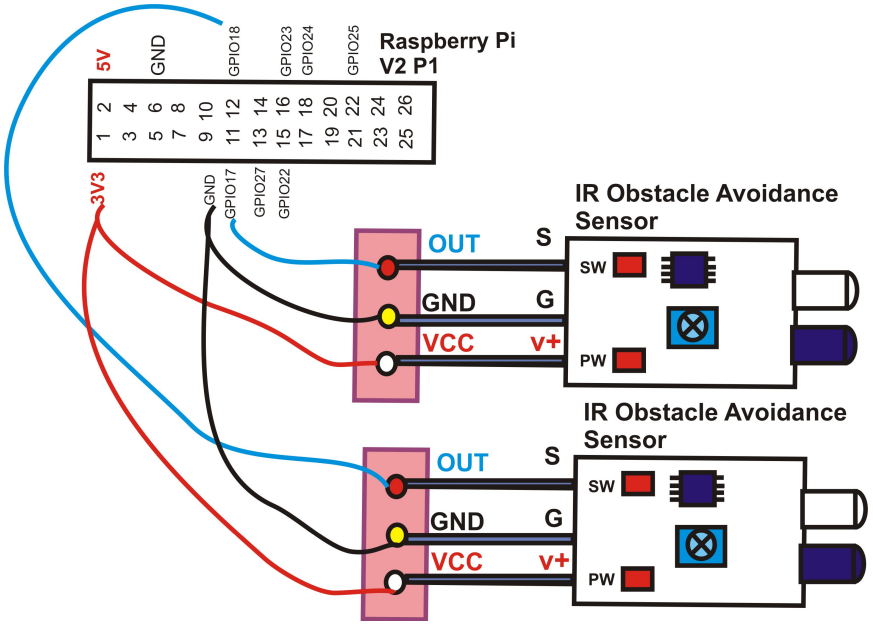
(2) broadcast "Update"

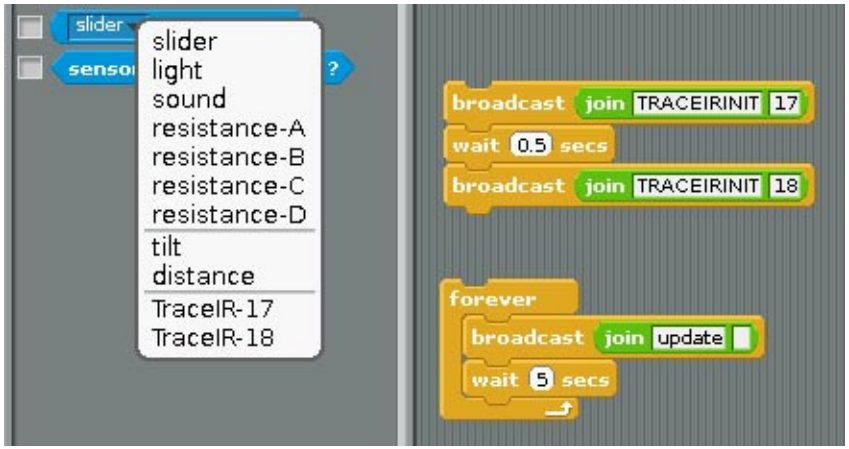
(3) in Sensing --> Slider , you will see the " I2C1B-0 ~ I2C1B-7" in the list



(6) IR Obstacle Avoidance Sensor module

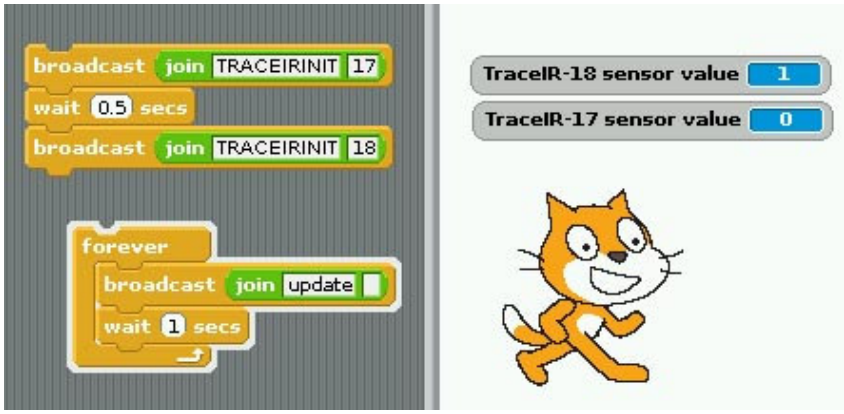




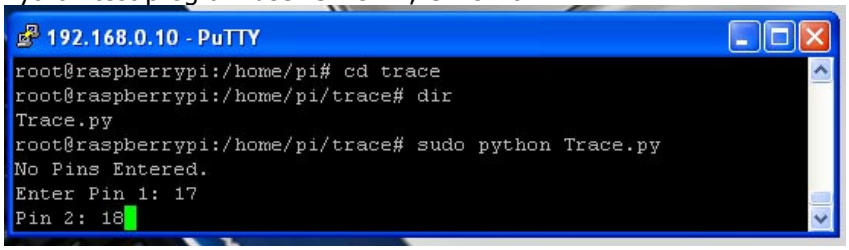


broadcast "TRACEIRINIT" +"17" GPIO number & "TRACEIRINIT" +"18"

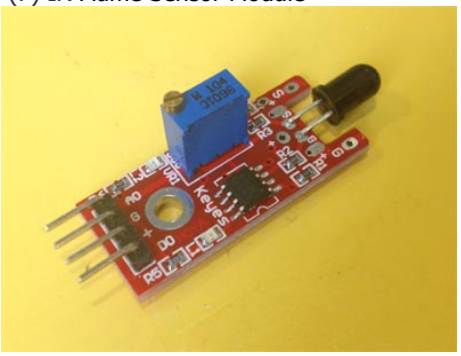
in Sensing --> Slider , you will see the "TraceIR-17" & "TraceIR-18" in the list



Python test program use GPIO 17, GPIO 18



(7) IR Flame Sensor Module



Comparator chip LM393. Detects a flame or a light source of a wavelength in the range of 760nm-1100 nm, Detection angle about 60 degrees, it is sensitive to the flame spectrum. Accuracy adjustable

Operating voltage 3.3V-5V

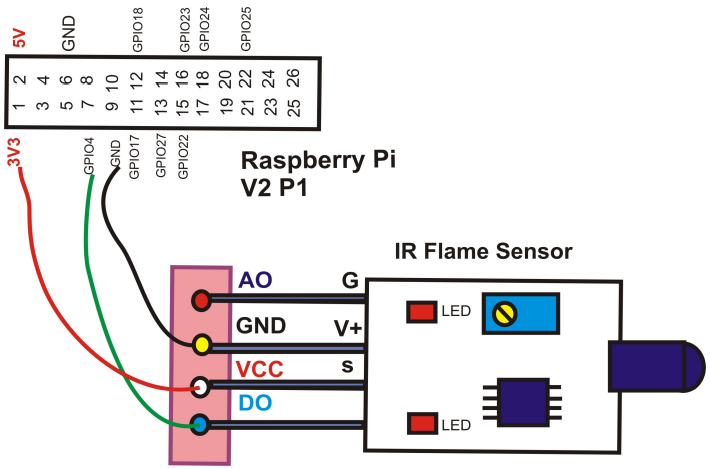
Output

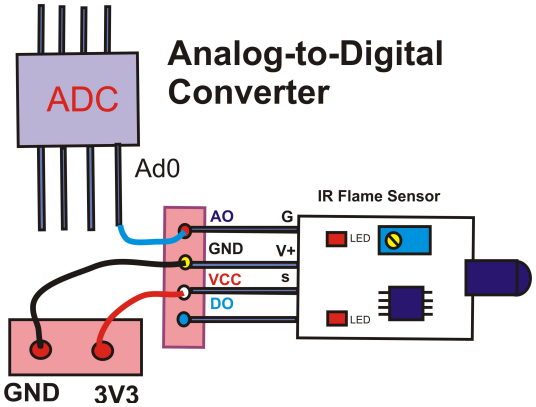
* DO digital switch outputs (0 and 1) * AO analogue voltage output
* power indicator and digital switch output indicator
* With a mounting screw hole * PCB size: 3.1cm x 1.5cm

Note: Please keep a distance with flame, high temperature maybe burn out the sensor module.

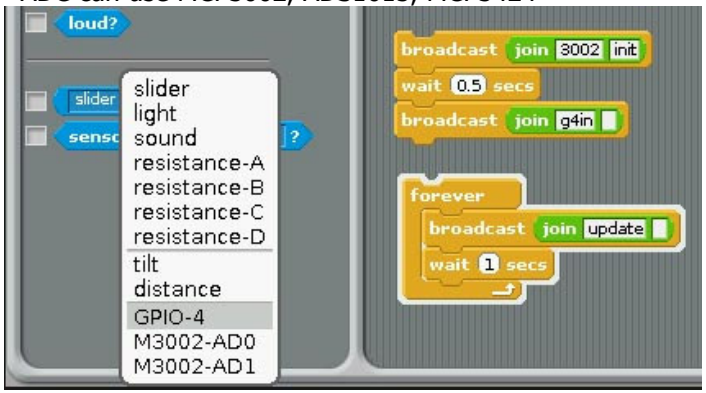
Interface Description (4-wire)

1 VCC: 3.3V-5V voltage
2 GND: GND
3 DO: board digital output interface (0 and 1)
4 AO: board analogue output interface



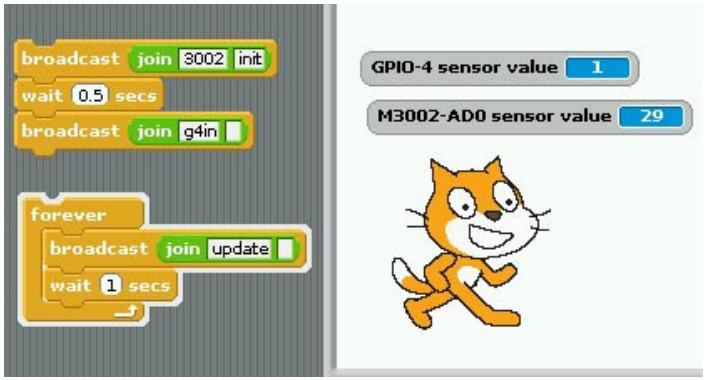


ADC can use MCP3002, ADS1015, MCP3424



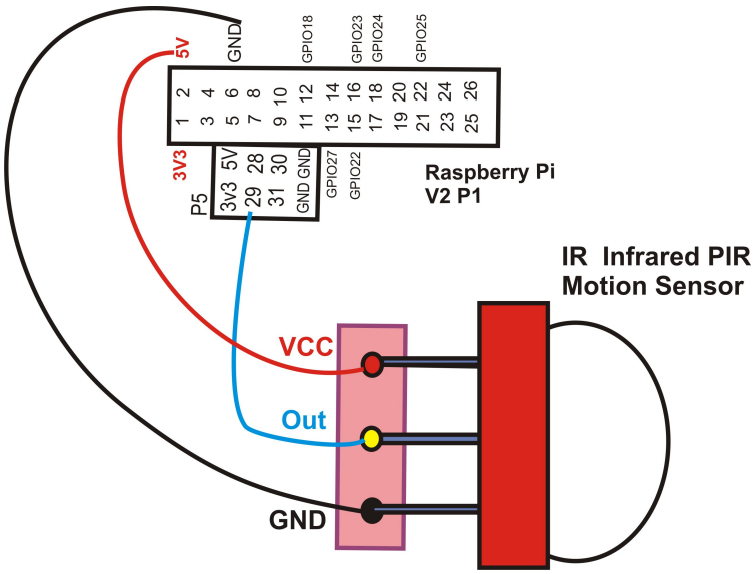
1.broadcast "3002init" & "g4in" GPIO 4 as input

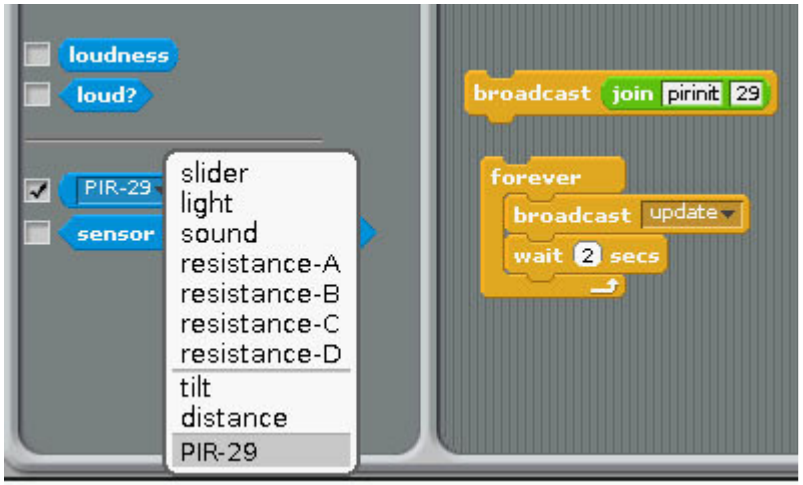
2.in Sensing --> Slider , you will see the "GPIO-4" & "M3002-AD0" & "M3002AD1" in the list



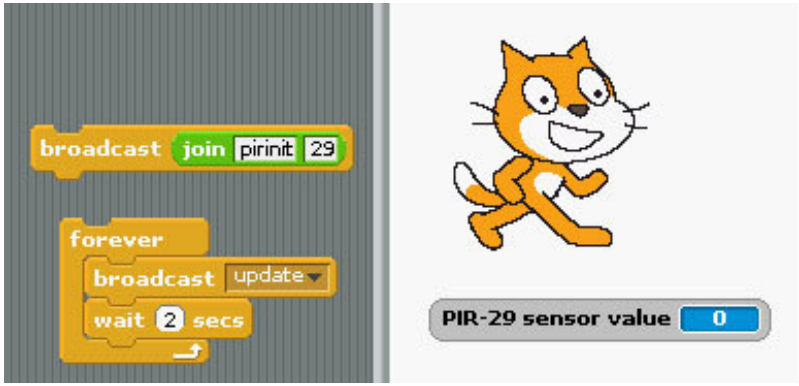
(8) IR PIR Motion Sensor Module

Pyroelectric PIR sensor, Fresnel
Low-power, static power 65uA
Wide voltage range, DC 5V-20V
Board small size 38 * 28mm (32 * 24 * mm)
Repeatable / not repeatable trigger mode selection
Easy to use, power + - signal output
7 m sensing range , 110 degree angle sensor
Electrical Parameters:
Operating voltage range of 5-20V DC voltage
65uA quiescent current, High output level 3.3 V / Low 0V
Non-repeatable trigger L trigger / H Repeat Trigger
PCB dimensions 38 * 28mm (32 * 24 * mm)



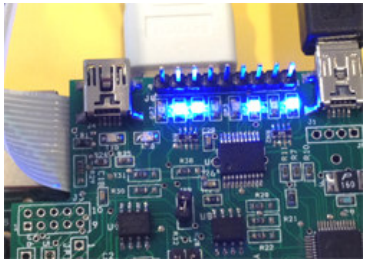


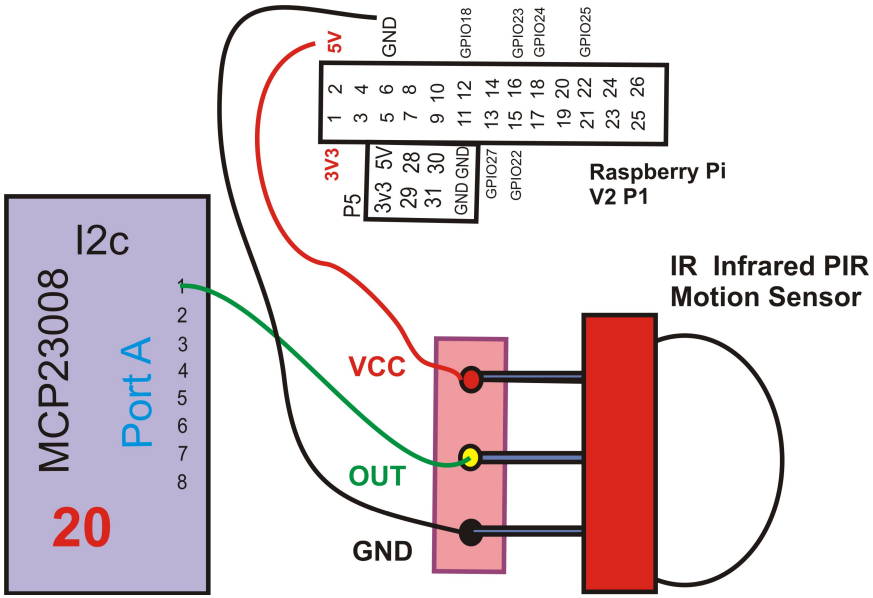
1.broadcast "pir" + "init" + "29" GPIO 29 as input
2.in Sensing --> Slider , you will see the "PIR-29" in the list

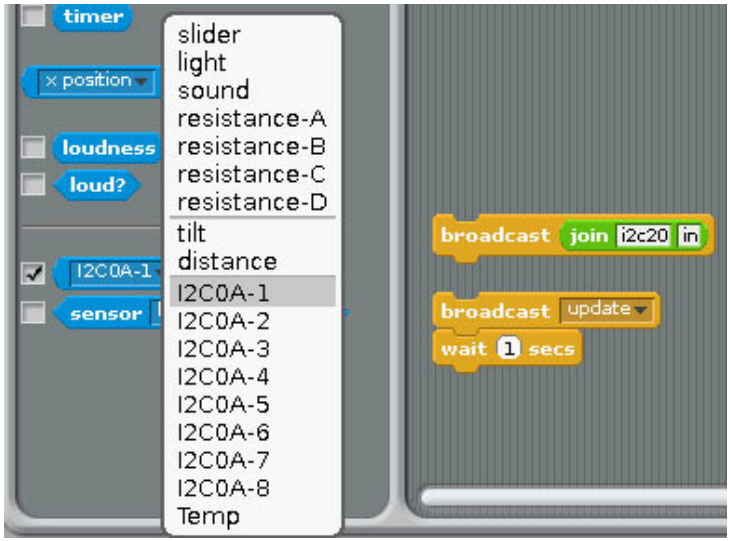


The PIR -29 value change between "0" & "1"

(8) i2c address 20 MCP23008 8 GPIO IR PIR Motion Sensor input test





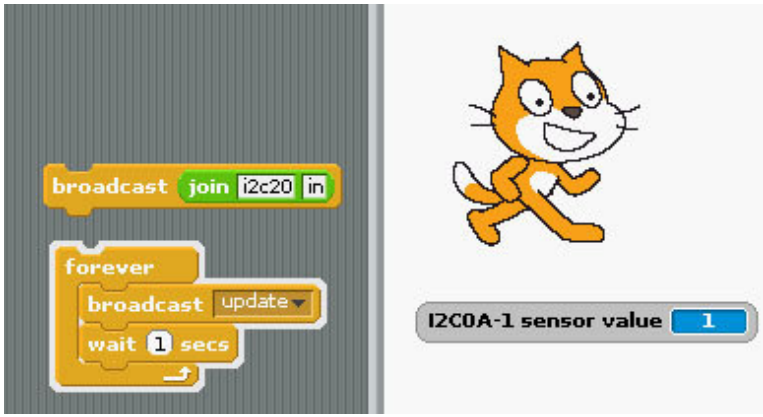


(1)"i2c20in" initial address 20, Port A as input

(2) broadcast "Update"

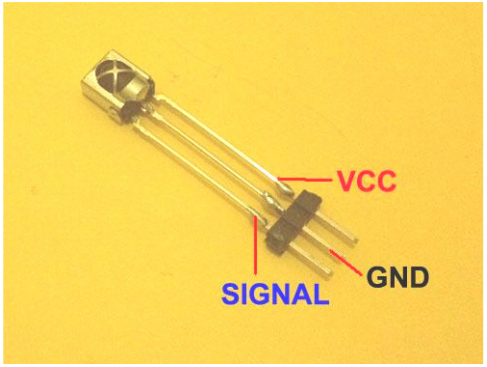
(3) in Sensing --> Slider , you will see the

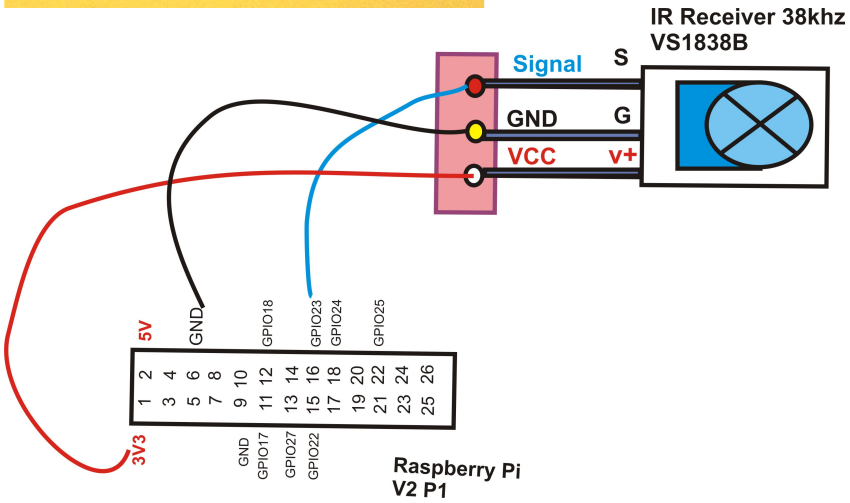
" I2C0A-1 ~ I2C0A-8" in the list



The I2C0A-1 value change between "0" & "1"

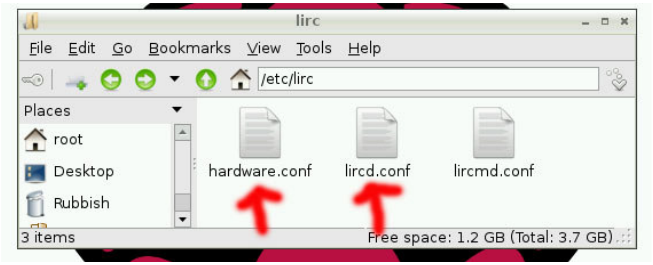
(9) IR remote control



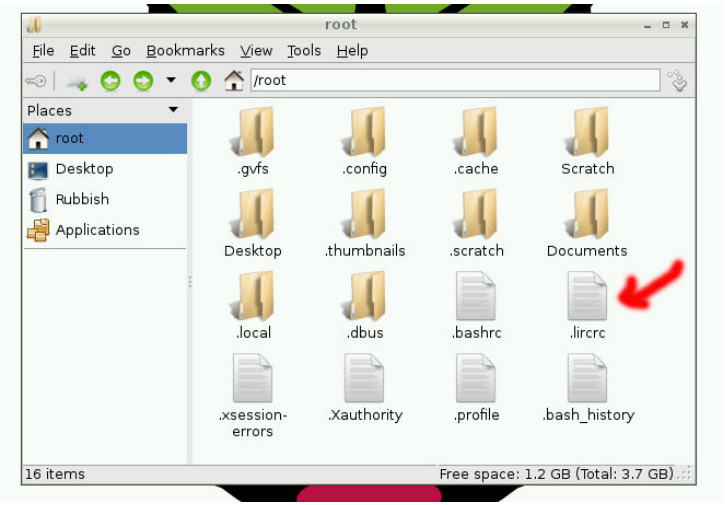


Download 3 Remote control setting files

lircd.conf .lircrc hardware.conf

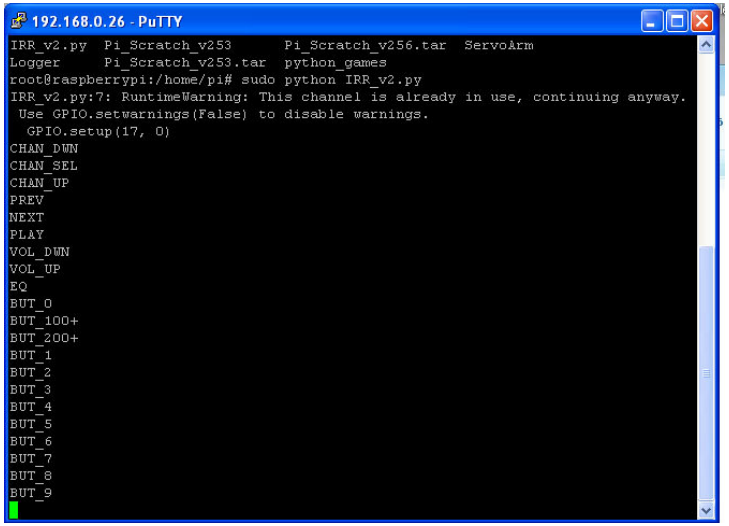


Copy hardware.conf & lircd.conf to /etc/lirc



Copy .lircrc to /root

Remote control example



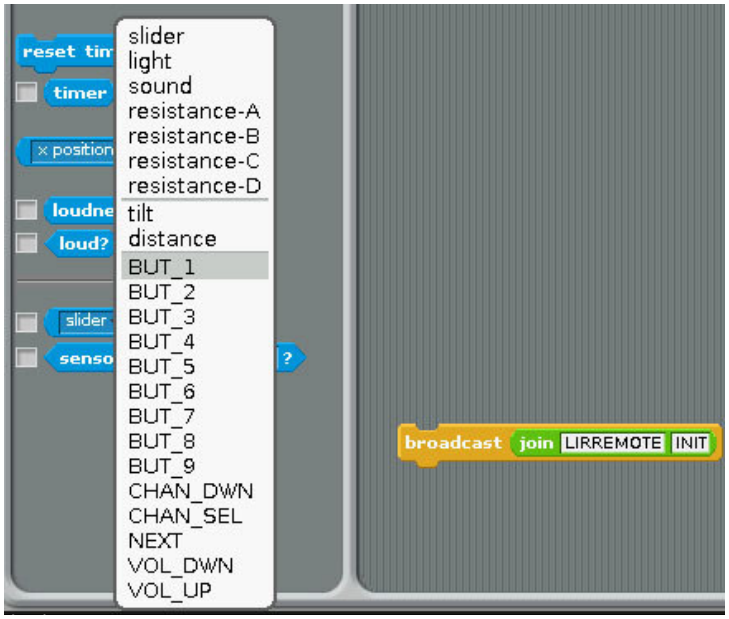
IRR_v2.py demo

```
192.168.0.26 - PuTTY
permitted by applicable law.
Last login: Tue Feb 25 14:23:24 2014
root@raspberrypi:~# irw
0000000000000001 00 BUT_CHANNEL_DOWN /etc/lirc/lircd.conf
0000000000000002 00 BUT_CHANNEL_SELECT /etc/lirc/lircd.conf
0000000000000003 00 BUT_CHANNEL_UP /etc/lirc/lircd.conf
0000000000000004 00 BUT_PREV /etc/lirc/lircd.conf
0000000000000005 00 BUT_NEXT /etc/lirc/lircd.conf
0000000000000006 00 BUT_PLAY /etc/lirc/lircd.conf
0000000000000007 00 BUT_VOL_DOWN /etc/lirc/lircd.conf
0000000000000008 00 BUT_VOL_UP /etc/lirc/lircd.conf
000000000000000a 00 BUT_0 /etc/lirc/lircd.conf
000000000000000b 00 BUT_100 /etc/lirc/lircd.conf
000000000000000c 00 BUT_200 /etc/lirc/lircd.conf
000000000000000d 00 BUT_1 /etc/lirc/lircd.conf
000000000000000e 00 BUT_2 /etc/lirc/lircd.conf
000000000000000f 00 BUT_3 /etc/lirc/lircd.conf
0000000000000010 00 BUT_4 /etc/lirc/lircd.conf
0000000000000011 00 BUT_5 /etc/lirc/lircd.conf
0000000000000012 00 BUT_6 /etc/lirc/lircd.conf
0000000000000013 00 BUT_7 /etc/lirc/lircd.conf
0000000000000014 00 BUT_8 /etc/lirc/lircd.conf
0000000000000015 00 BUT_9 /etc/lirc/lircd.conf
```

irw read information from remote control

```
192.168.0.26 - PuTTY
root@raspberrypi:~# irexec
Channel Down pressed
Channel SELECT pressed
Channel UP pressed
PREV pressed
NEXT pressed
PLAY pressed
VOL DOWN pressed
VOL UP pressed
EQ pressed
0 pressed
100+ pressed
200+ pressed
1 pressed
2 pressed
3 pressed
4 pressed
5 pressed
6 pressed
7 pressed
8 pressed
9 pressed
```

irexec read information from remote control

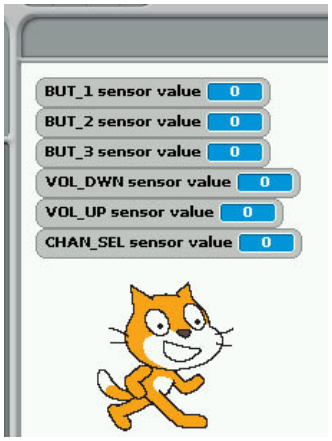


Scratch setting demo

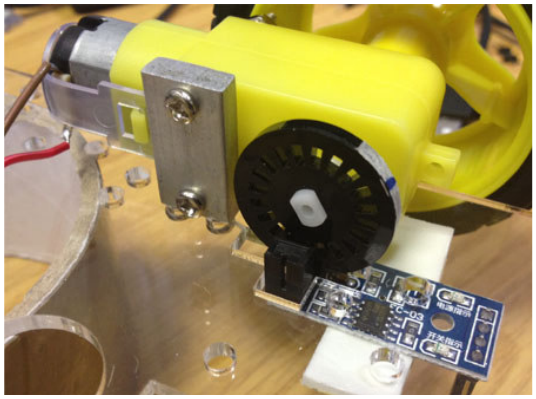
Command " LIRREMOTE" +"INIT" to active IR REMOTE CONTROL

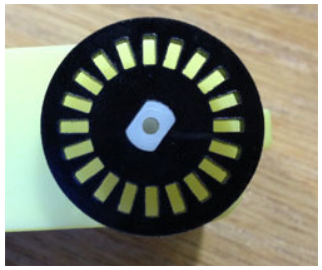
1.broadcast "LIRREMOTEINIT"

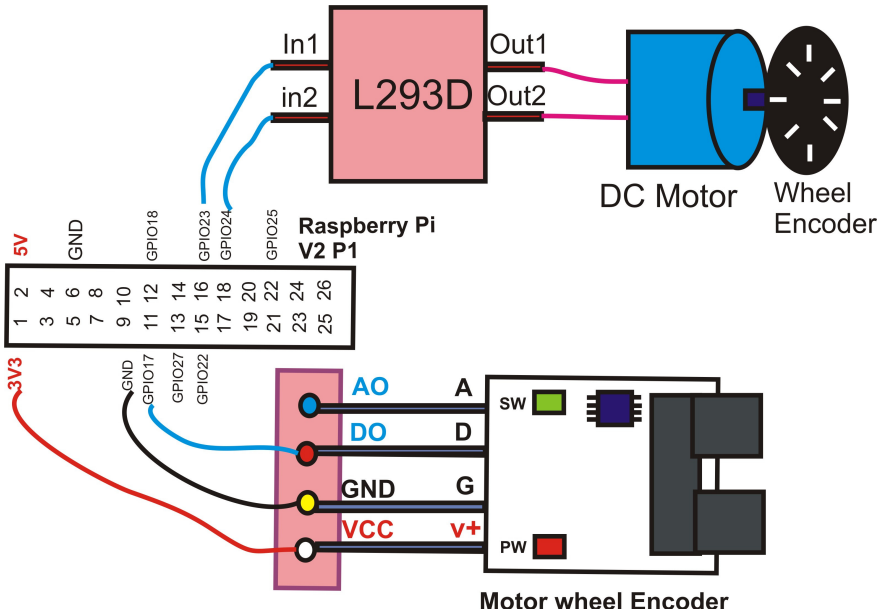
2. after broadcast Press the key in remote control you want use in Scratch in Sensing --> Slider , you will see the key you press in the list



(10) Motor Wheel Encoder



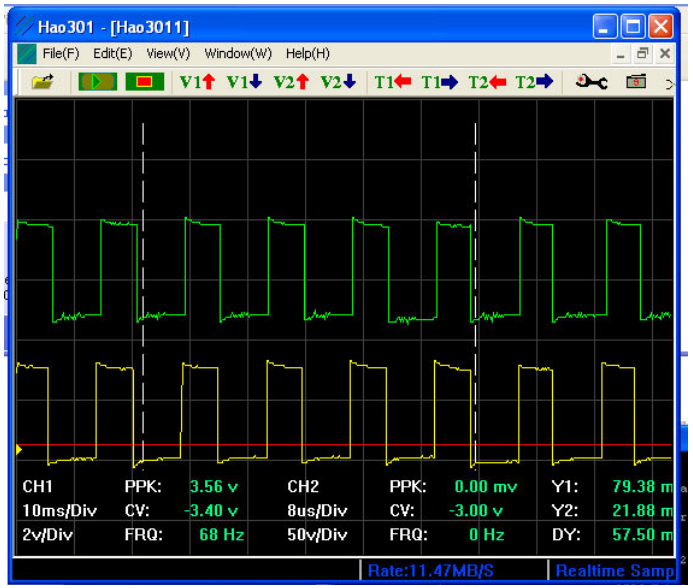




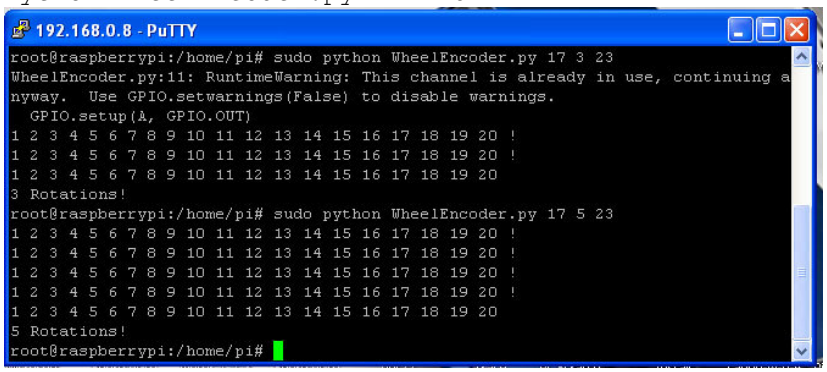
WheelEncoder.py Python command

Sudo python WheelEncoder.py 17 3 23

17 GPIO for Encoder 3 Rotation
23 Motor connect to GPIO 23

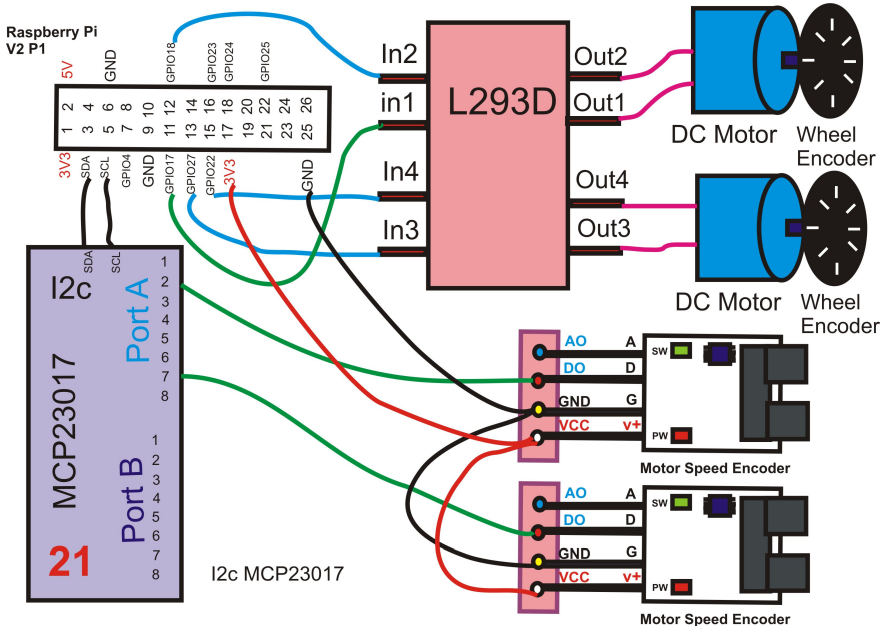


Python WheelEncoder.py DEMO



Command "23" + "MOTORDIS" "17"+ "S" + "3"
17 GPIO for Encoder 3 Rotation
23 Motor connect to GPIO 23

(11) i2c address 21 MCP23017 16 GPIO Wheel Encoder

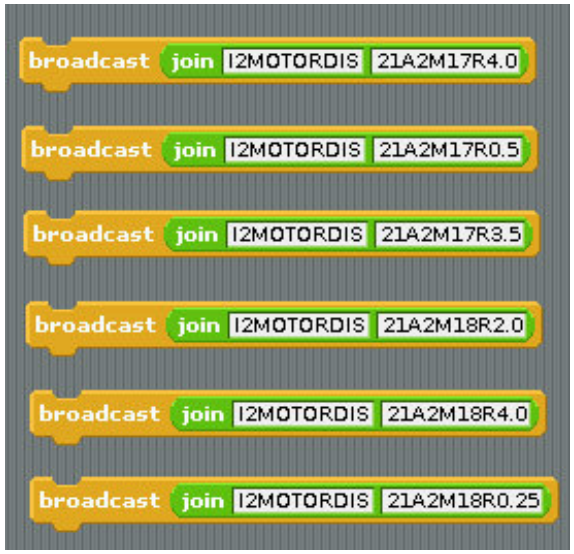


Command

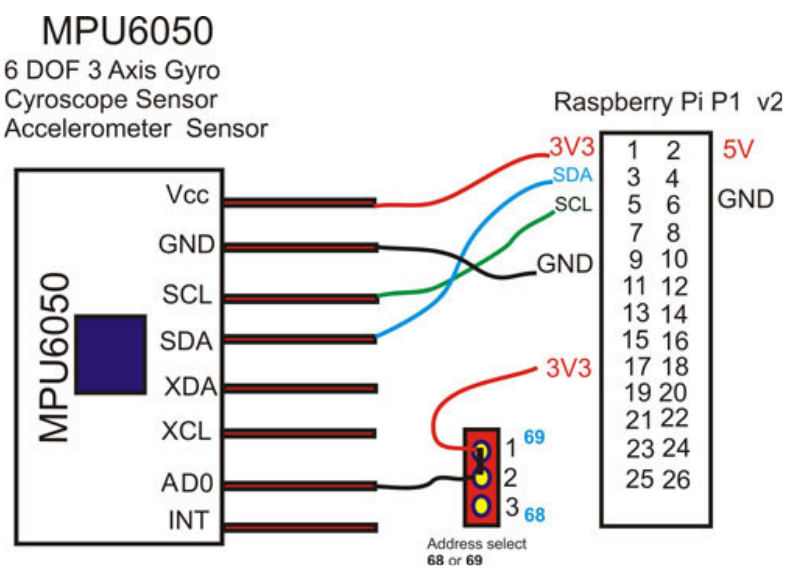
I2MOTORDIS[Address][Bank][Encoder Pin]M[Motor Pin]R[Rotations]

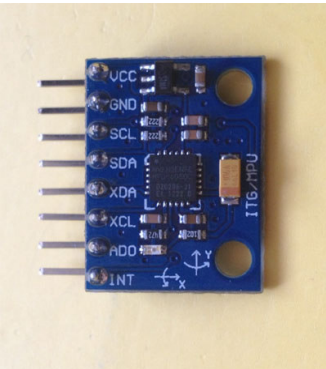
I2MOTORDIS**21**A**2**M**17**R**2**

1. Address = **20 - 27**, Address for the I2C
2. DeviceBank = **A** or **B** (Bank for the encoder)
3. Encoder Pin = **1 - 8** (Pin number for the encoder)
4. Motor Pin = GPIO Pin number for the Motor
For example our Motor Board use GPIO **17** & **18** control 2 Motor forward & backward
5. Rotations = How many rotations you want the Motor to do before it stops. Minimum input 0.2



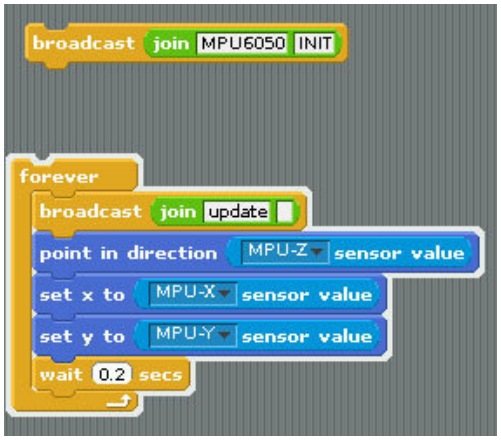
(12) i2c address 69 MPU6050 3 Axis Gyroscope + Accelerometer Module

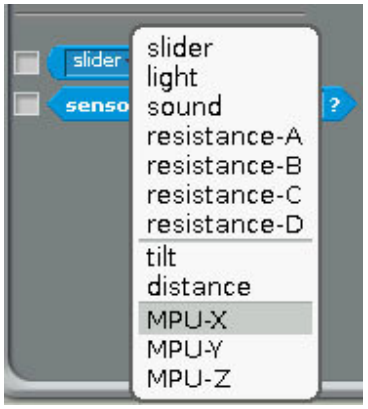




The MPU-6050 sensor contains a MEMS accelerometer and a MEMS gyro in a single chip. It is very accurate, since it contains 16-bits analogue to digital conversion hardware for each channel. Therefore it captures the x, y, and z channel at the same time. The MPU-6050 is not expensive, since it combines both an accelerometer and a gyro.

Features
Name: MPU-6050 module (three axis gyro + three axis acceleration)
Use chip: MPU-6050
The power supply: 3-5 v (internal low voltage difference voltage stabilizing)
Communication: standard IIC communication agreement
Chip built-in 16 bit AD converter, 16 bits of data output
The gyroscope range: + 250 500 1000 2000 ° /s
Acceleration range: ± 2 ± 4 ± 8 ± 16 g



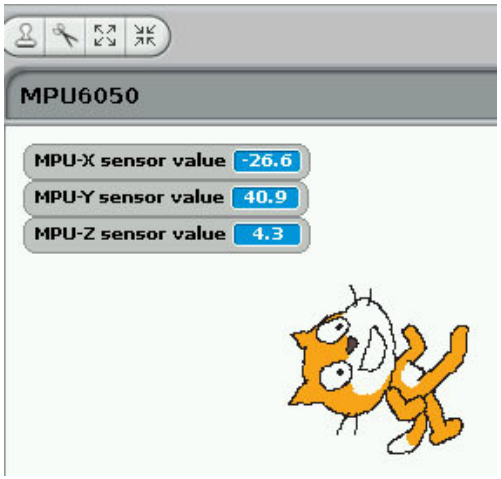


Command " MPU6050" +"INIT" to active MPU6050

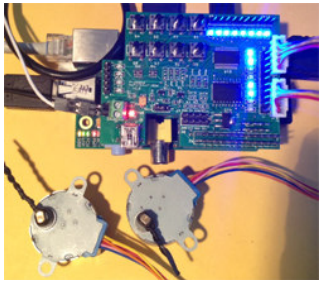
1.broadcast "MPU6050INIT"

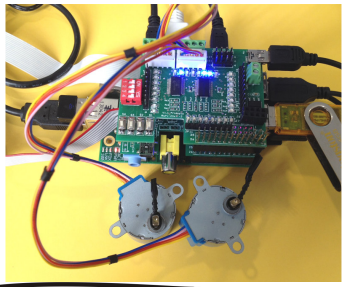
2. broadcast "update"

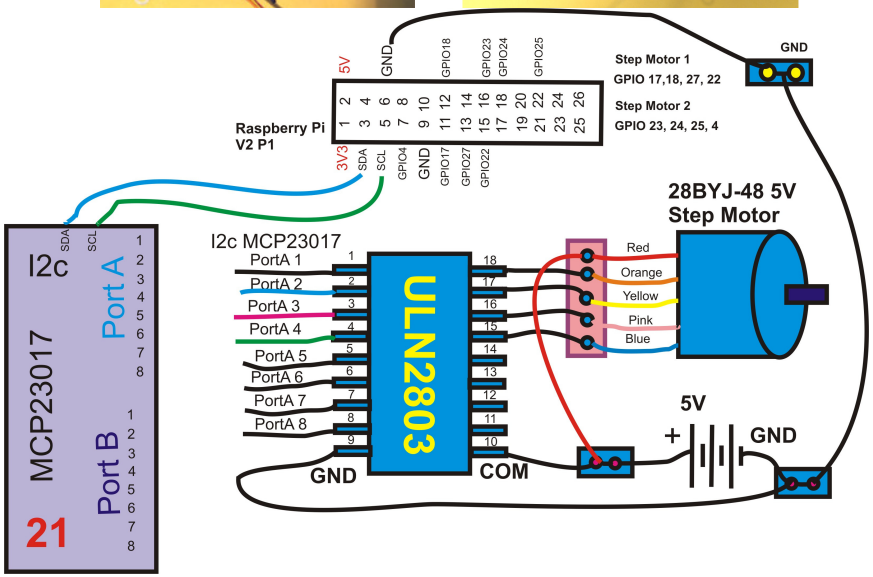
3. in Sensing --> Slider , you will see the " MPU-X, MPU-Y, MPU-Z" in the list



26. Demo 21 i2c address 20 MCP23017 16 GPIO Step Motor
smotor command







command "i2smotor " + "address" + "port" +"mode" + "Step"

address 20 ,21 ,22, 23, 24, 25, 26, 27

port a or b (Port A or Port B)

mode 1 first 4 bit Step Motor A

2 Step Motor B

3 A & B move together A clockwise , B anticlockwise

4 A & B move together A & B clockwise or anticlockwise

Step 500 **anticlockwise 500 step**

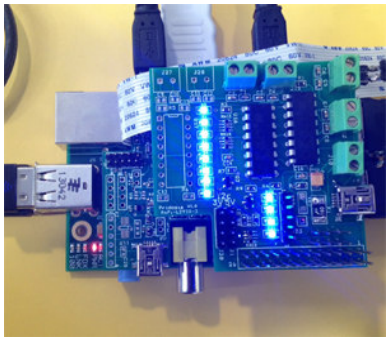
-500 **clockwise 500 Step**

Full command i2smotor21a1500

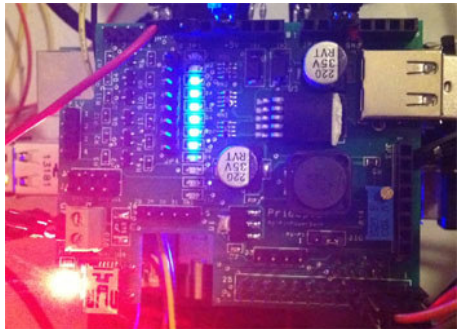
Address 21, port A ,Step motor A, anticlockwise, 500 step

27. Demo 22 GPIO LED control

8 + 4 LED control



4 Motor P5 Board



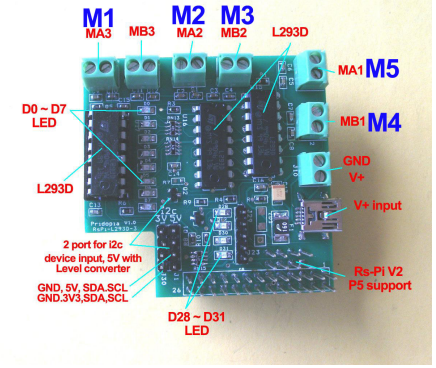
PSU Board

Command
GPWMLED"A" or "B" or "C" or "D"

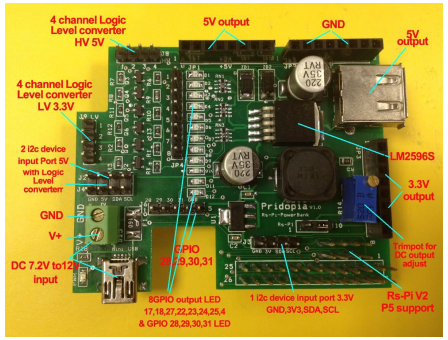
A GPIO 17,18,27,22,23,24,25,4
B GPIO 28,29,30,31
C GPIO A + B
Z STOP LED flashing

S"speed 1 to 10" 1 fast 10 slow"

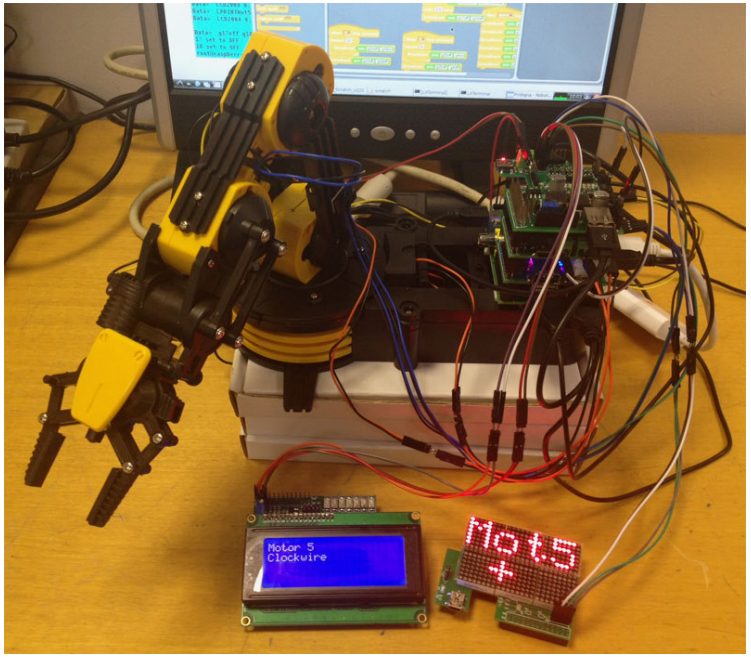
28. Demo 23 Maplin Robotic ARM control



Motor Board



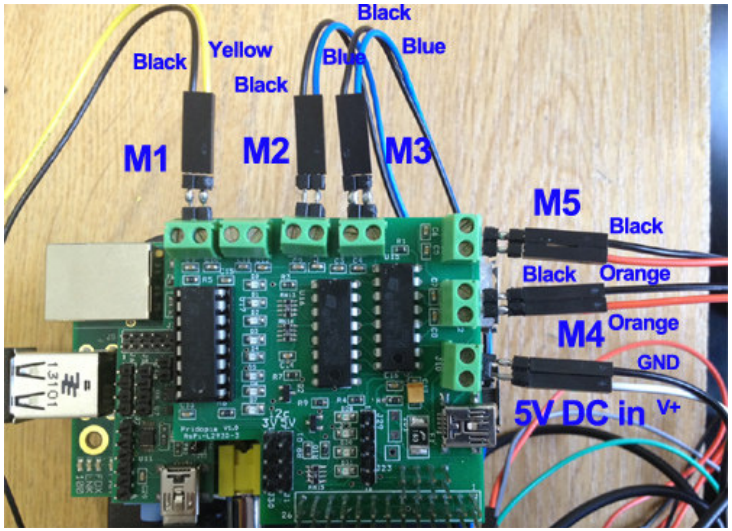
Power Board

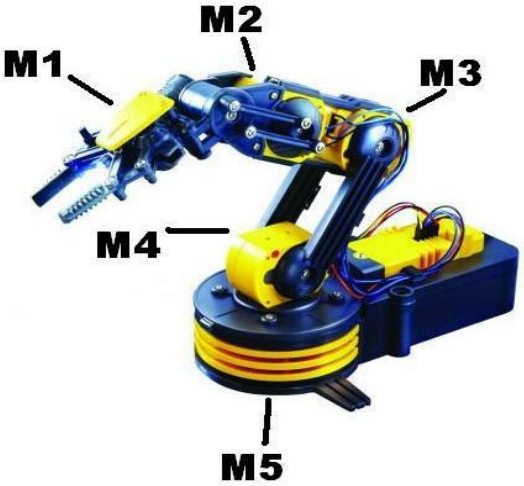


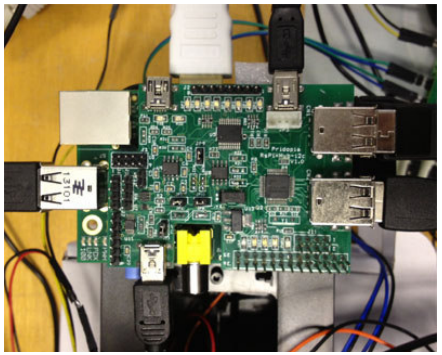
Use 1.Rs-pi-L293D-3 6 Motor Board -
Control Robot Arm 5 Motor (M1,M2,M3,M4,M5)
2.Rs-Pi-Power Bank Multi-function Board
Use DC12V adapter, provide Raspberry Pi 5V and all the power for 5 DC Motor
3. Rs-Pi-4hub Hub i2c multi function Board
Provide extra USB Hub & extra i2c function

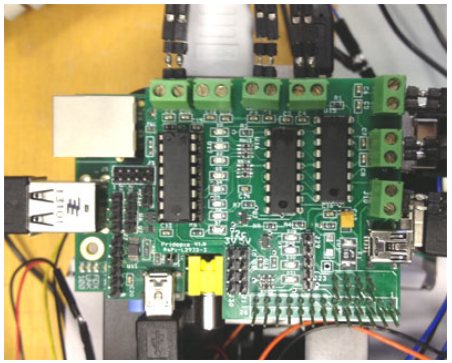
(option: display information)

1. Rs-Pi- i2c LCD2004 Master Board
2. Rs-Pi-24x16 LED Matrix



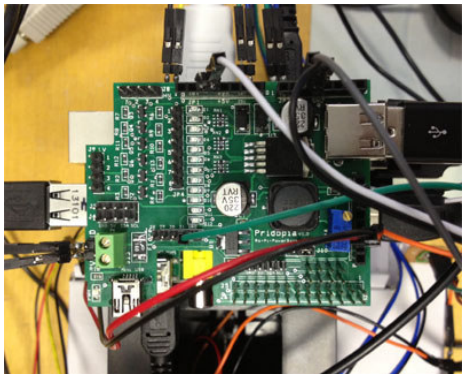


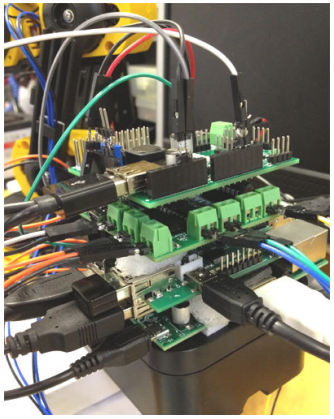




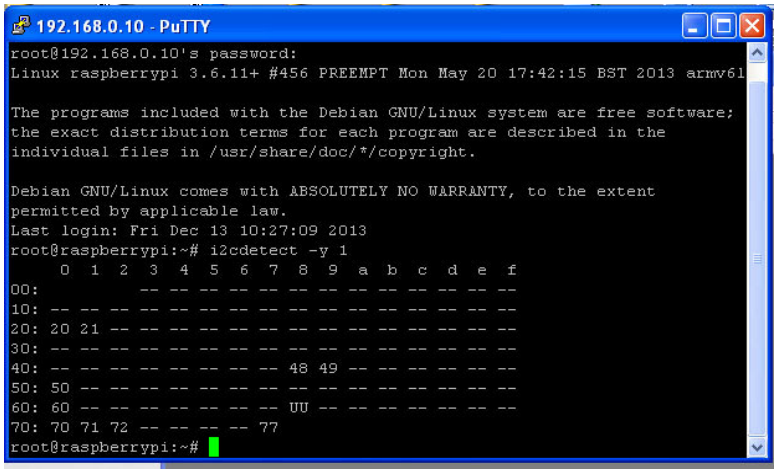
First Layer 4 USB hub i2c multi function board

Second Layer 6 Motor Board



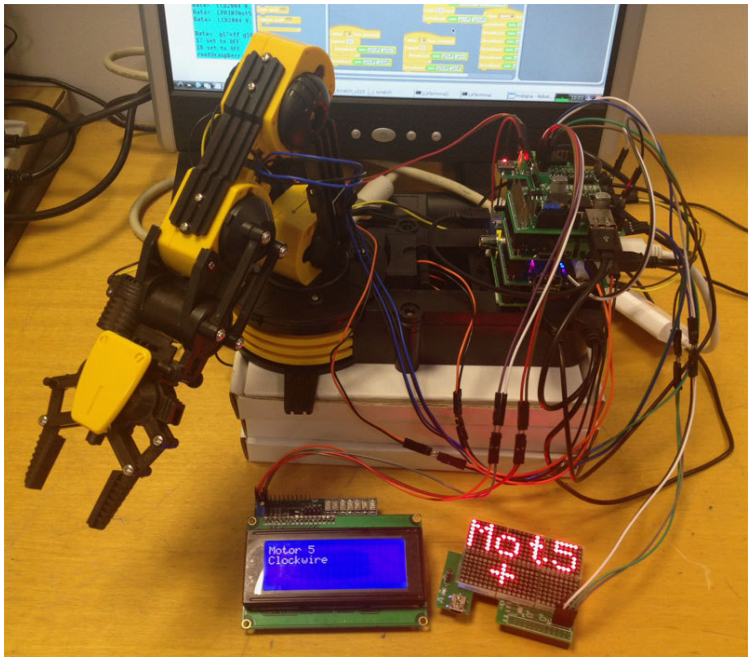


Third Layer Power Bank Board

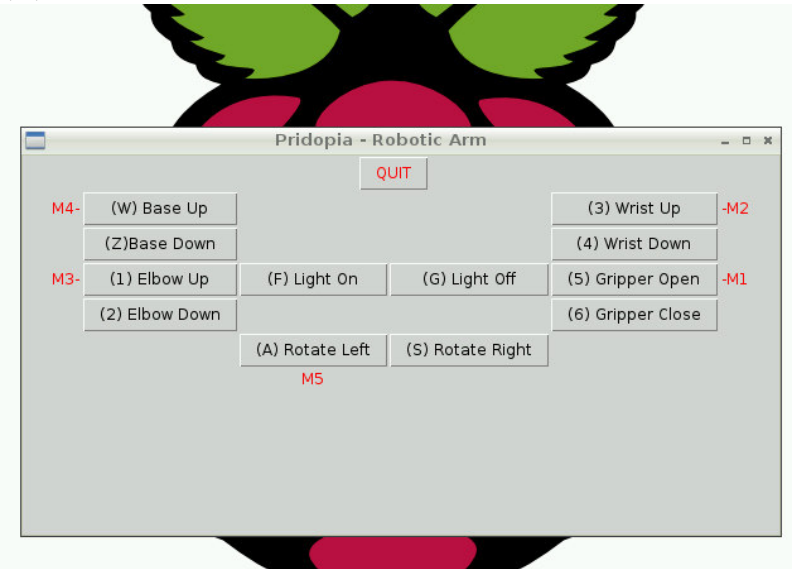


I2cdevice list

20 -> MCP23008 48 -> TMP102 68 -> RTC DS1307
50 -> 24c32 60 -> MCP4725 49-> ADS1015
21 → MCP23017 i2c LCD2004 Master Board
70,71,72 → i2c 24x16 LED Matrix
77 → BMOP085



(1) xWindows control mode

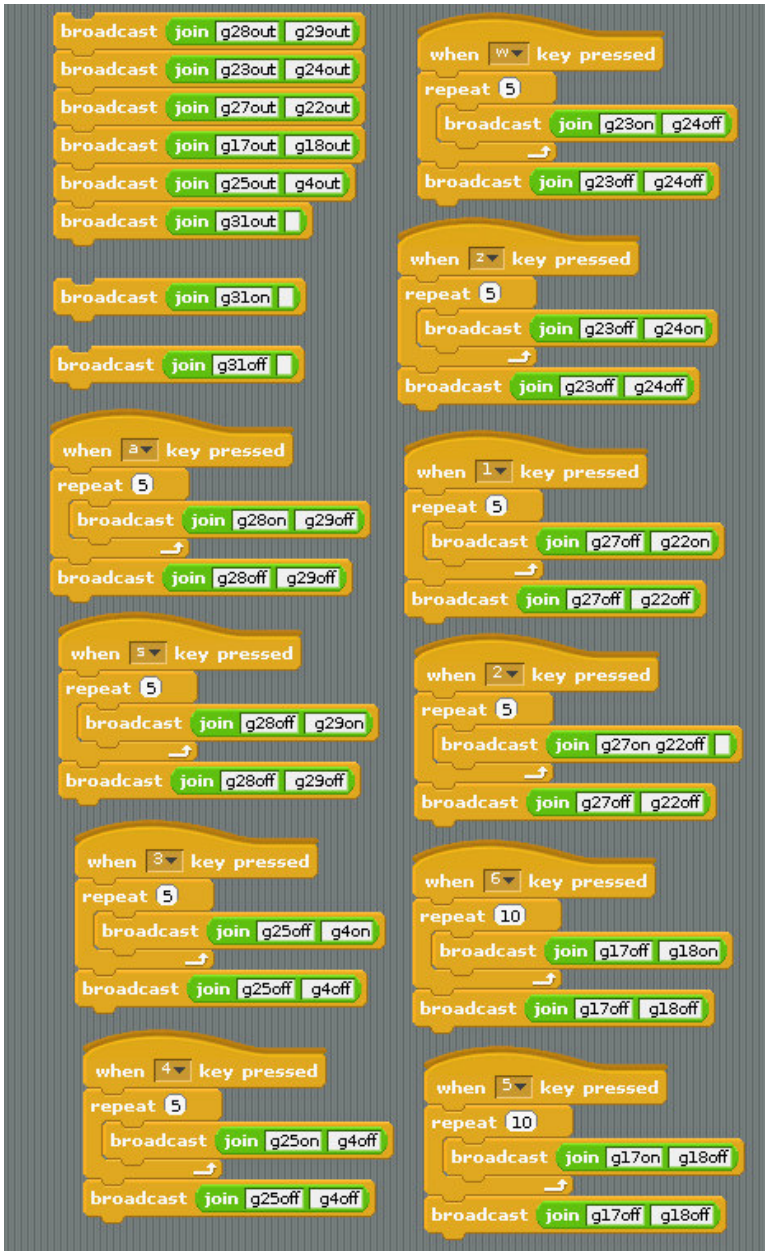


Use mouse & keyboard control in xWindows
control 5 Motor (M1, M2, M3, M4, M5) and LED ON/OFF

(2)Scratch GPIO control mode

1. setting GPIO 17,18,27,22,23,24,25,4 as output
2. setting GPIO 28,29,31 as output

```
a KEY GPIO 28,29  M1   CLOCKWISE
s KEY GPIO 28,29  M1   ANTICLOCKWISE

w KEY GPIO 23,24  M2   CLOCKWISE
z KEY GPIO 23,24  M2   ANTICLOCKWISE

1 KEY GPIO 27,22  M4   CLOCKWISE
2 KEY GPIO 27,22  M4   ANTICLOCKWISE

3 KEY GPIO 25,4   M3   CLOCKWISE
4 KEY GPIO 25,4   M3   ANTICLOCKWISE

5 KEY GPIO 17,18  M5   CLOCKWISE
6 KEY GPIO 17,18  M5   ANTICLOCKWISE
```



(3)Scratch PWM command mode

Include i2c 20x4 LCD Screen & i2c 24x16 LED Matrix

A - GPIO 17,18 Motor A B - GPIO 27,22 Motor B

C - GPIO 23,24 Motor C D - GPIO 25,4 Motor D

E - GPIO 28,29 Motor E

command "Motor Name"+ "DM"+"speed"

speed (10 ~100) clockwise

speed (-10 ~ -100) anticlockwise

```
6 KEY GPIO 28,29  M1   CLOCKWISE
5 KEY GPIO 28,29  M1   ANTICLOCKWISE

3 KEY GPIO 23,24  M2   CLOCKWISE
4 KEY GPIO 23,24  M2   ANTICLOCKWISE

w KEY GPIO 27,22  M4   CLOCKWISE
z KEY GPIO 27,22  M4   ANTICLOCKWISE

2 KEY GPIO 25,4   M3   CLOCKWISE
1 KEY GPIO 25,4   M3   ANTICLOCKWISE

a KEY GPIO 17,18  M5   CLOCKWISE
s KEY GPIO 17,18  M5   ANTICLOCKWISE
```

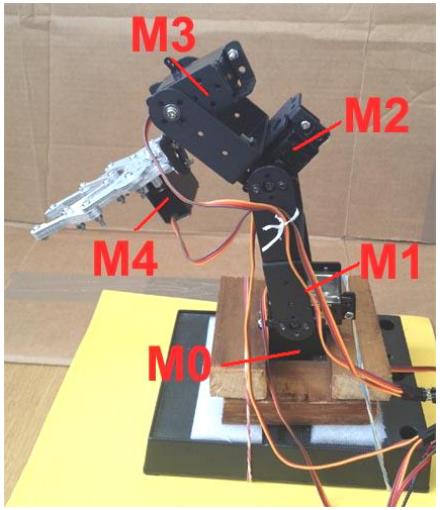
```
broadcast join LCD2004 init
wait 0.3 secs
broadcast join LCD2004 bl on

broadcast join LED CLS

broadcast join LCD2004 cls

broadcast join LPRINTPWM TEST

broadcast join LCD2004 bl off

broadcast join LCD2004 rss 6

broadcast join LCD2004 rss off

when 1 key pressed
repeat 20
  broadcast join DDM 100
broadcast join LCD2004 cls
wait 0.2 secs
broadcast join DDM 0
broadcast join LCD2004 Motor 3
wait 0.3 secs
broadcast join LCD2004 0,1 ClockWise
wait 0.3 secs
broadcast join LPRINT M3 --

when 2 key pressed
repeat 20
  broadcast join DDM -100
broadcast join LCD2004 cls
wait 0.2 secs
broadcast join DDM 0
broadcast join LCD2004 Motor 3
wait 0.3 secs
broadcast join LCD2004 0,1 ClockWise
wait 0.3 secs
broadcast join LPRINT M3 ++

when w key pressed
repeat 20
  broadcast join BDM 100
broadcast join LCD2004 cls
wait 0.2 secs
broadcast join BDM 0
broadcast join LCD2004 Motor 4
wait 0.3 secs
broadcast join LCD2004 0,1 ClockWise
wait 0.3 secs
broadcast join LPRINT M4 ++

when z key pressed
repeat 10
  broadcast join BDM -100
broadcast join LCD2004 cls
wait 0.2 secs
broadcast join BDM 0
broadcast join LCD2004 Motor 4
wait 0.3 secs
broadcast join LCD2004 0,1 AntiClockWise
wait 0.3 secs
broadcast join LPRINT M4 --
```

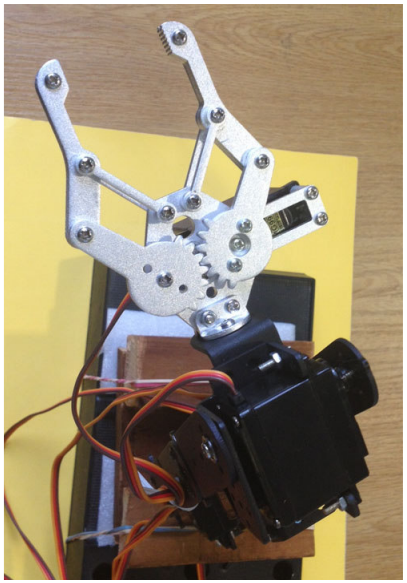


```
when s key pressed
repeat 20
  broadcast join ADM 100
broadcast join LCD2004 cls
wait 0.2 secs
broadcast join ADM 0
broadcast join LCD2004 Motor 5
wait 0.3 secs
broadcast join LCD2004 0,1 AntiClockWise
wait 0.3 secs
broadcast join LPRINT M5 --

when a key pressed
repeat 20
  broadcast join ADM -100
broadcast join LCD2004 cls
wait 0.2 secs
broadcast join ADM 0
broadcast join LCD2004 Motor 5
wait 0.3 secs
broadcast join LCD2004 0,1 ClockWise
wait 0.3 secs
broadcast join LPRINT M5 ++

when 4 key pressed
repeat 20
  broadcast join CDM 100
broadcast join LCD2004 cls
wait 0.2 secs
broadcast join CDM 0
broadcast join LCD2004 Motor 2
wait 0.3 secs
broadcast join LCD2004 0,1 AntiClockWise
wait 0.3 secs
broadcast join LPRINT M2 --

when 5 key pressed
repeat 20
  broadcast join EDM 100
broadcast join LCD2004 cls
wait 0.2 secs
broadcast join EDM 0
broadcast join LCD2004 Motor 1
wait 0.3 secs
broadcast join LCD2004 0,1 AntiClockWise
wait 0.3 secs
broadcast join LPRINT M1 --

when 6 key pressed
repeat 20
  broadcast join EDM -100
broadcast join LCD2004 cls
wait 0.2 secs
broadcast join EDM 0
broadcast join LCD2004 Motor 1
wait 0.3 secs
broadcast join LCD2004 0,1 ClockWise
wait 0.3 secs
broadcast join LPRINT M1 ++

when 3 key pressed
repeat 20
  broadcast join CDM -100
broadcast join LCD2004 cls
wait 0.2 secs
broadcast join CDM 0
broadcast join LCD2004 Motor 2
wait 0.3 secs
broadcast join LCD2004 0,1 ClockWise
wait 0.3 secs
broadcast join LPRINT M2 ++
```

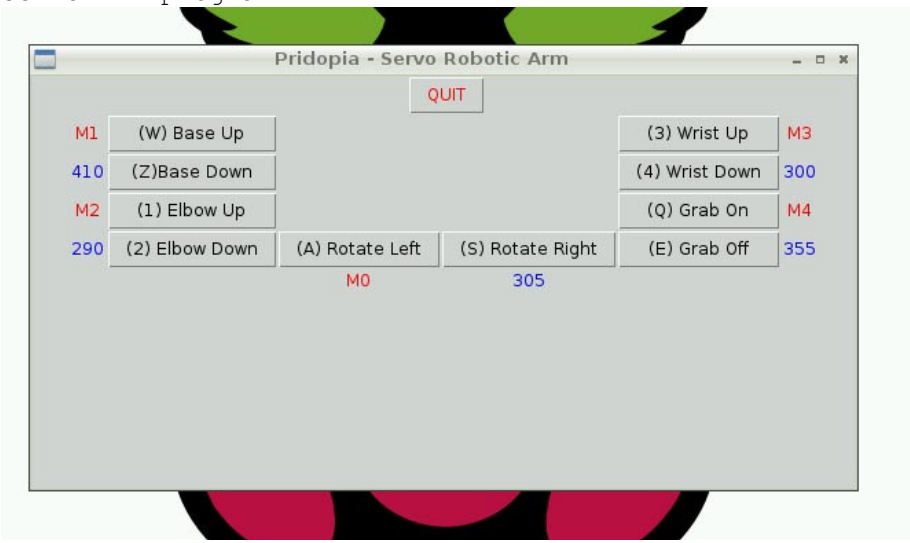
29. Demo 24 Servo Motor Robot ARM control

6 DOF Manipulator Aluminum Robot Arm

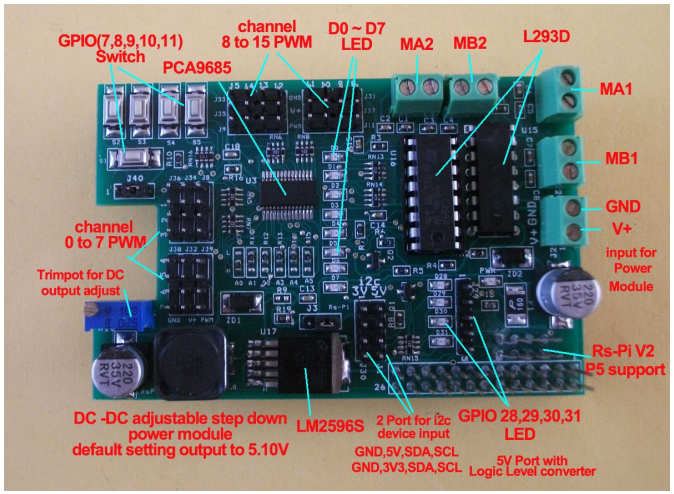


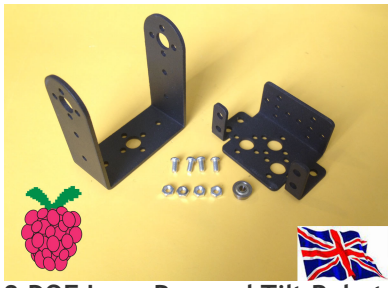


Servo-Arm program

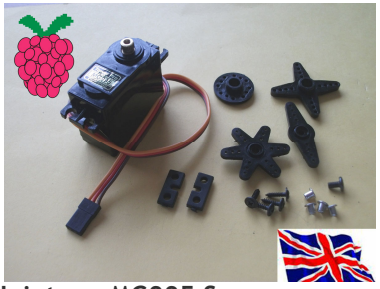


Servo Arm control in Scratch





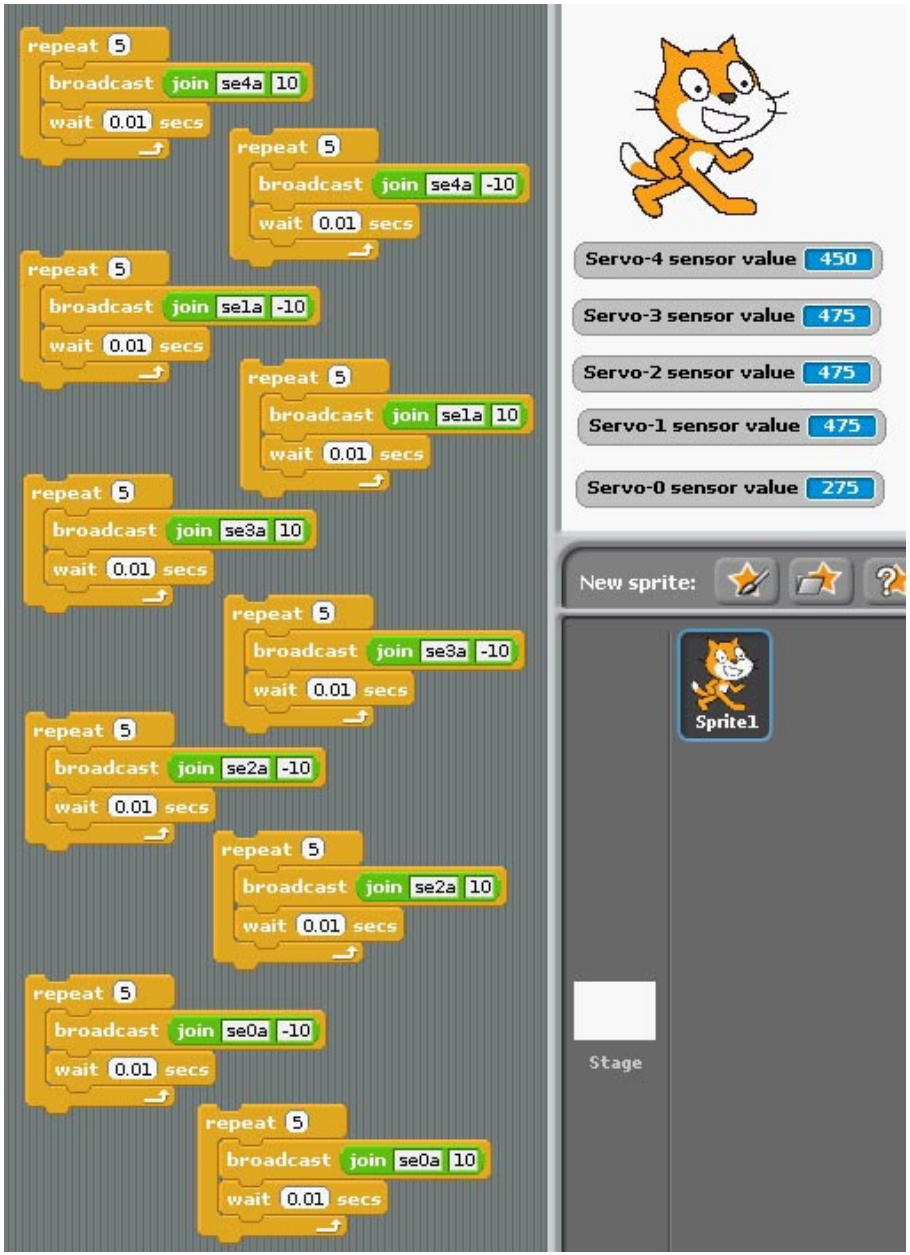
2 DOF Long Pan and Tilt Robot Joint



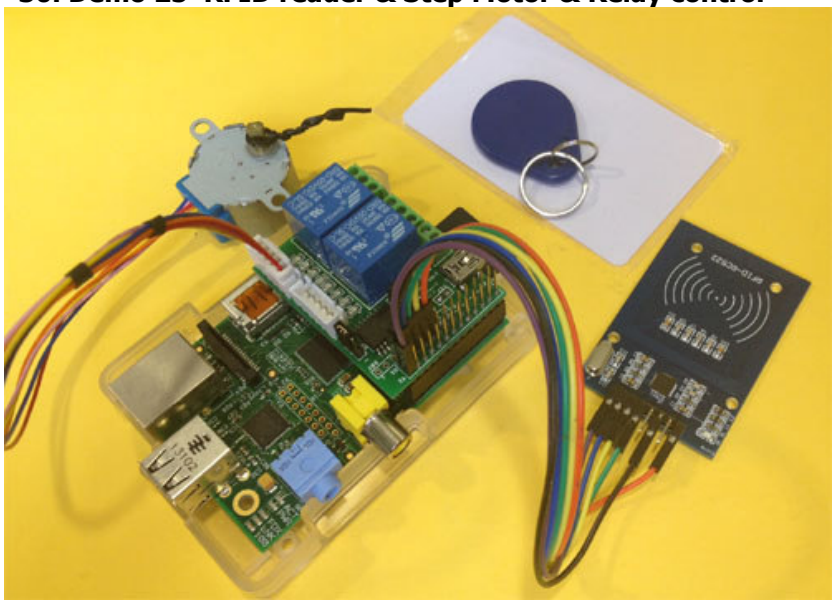
MG995 Servo

3 set x 2 DOF Long Pan and Tilt Robot Joint
5 x MG995 Servo Motor
1 x Gripper set

```
Use servo channel 0,1,2,3,4,   connect Servo 0 to 4

Command "SE"+ "PWM (0-15)" + "a" +"angle"  for Address 41
    Se4a10   --> channel 4 servo move 10 angle  address 41
    Se4a-10  --> channel 4 servo move -10 angle address 41
    se0a10   --> channel 0 servo move 10 angle  address 41
    se0a-10  --> channel 0 servo move -10 angle  address 41
```



30. Demo 25 RFID reader & Step Motor & Relay control

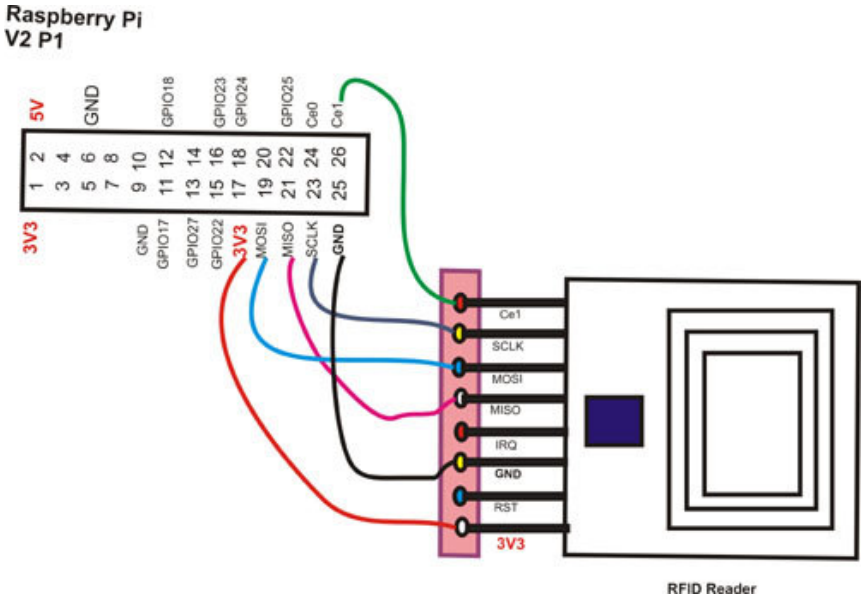


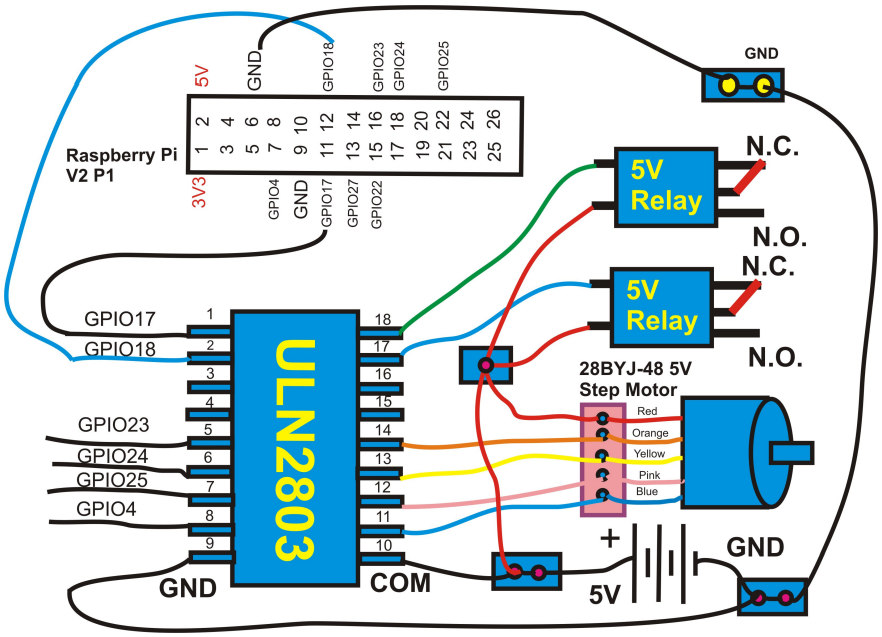
Install tools for RFID kit

In folder" Installer"

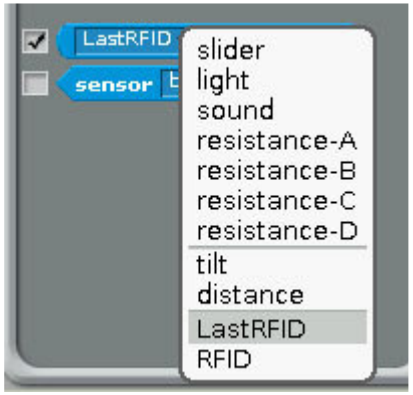
Sudo python RFID-Installer.py

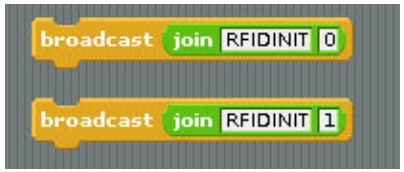
(1) GPIO (SPi) RFID Reader & GPIO Relay output





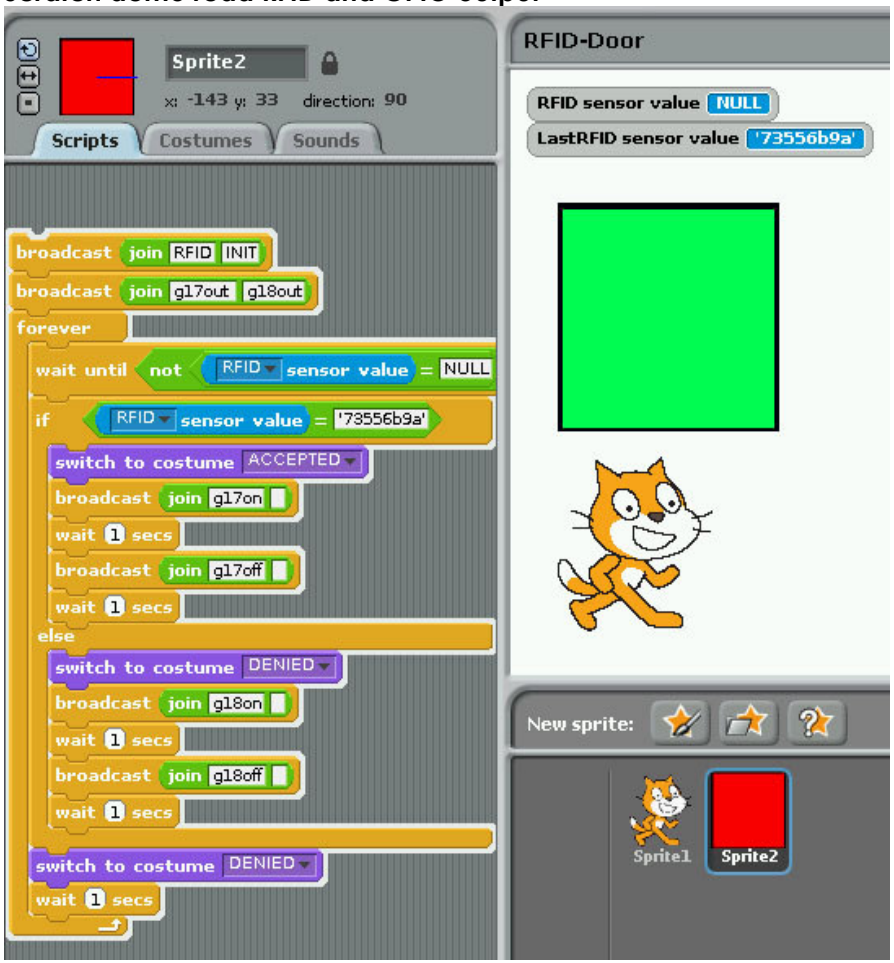
Scratch control demo





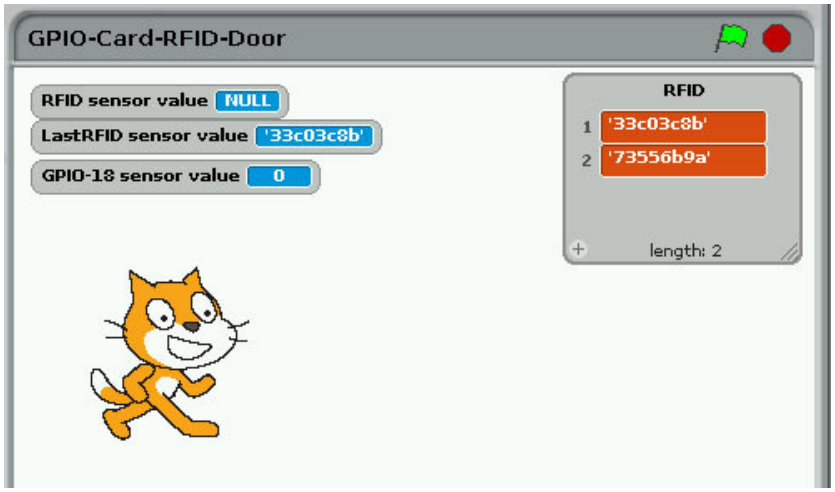

```
1) Command "RFID"+"INIT"+"0"  for CE0
   or Command "RFID"+"INIT"+"1"  for CE1

will initial SPI signal to active RFID Reader
2) you will see "LastRFID" & "RFID" in Sensors
```

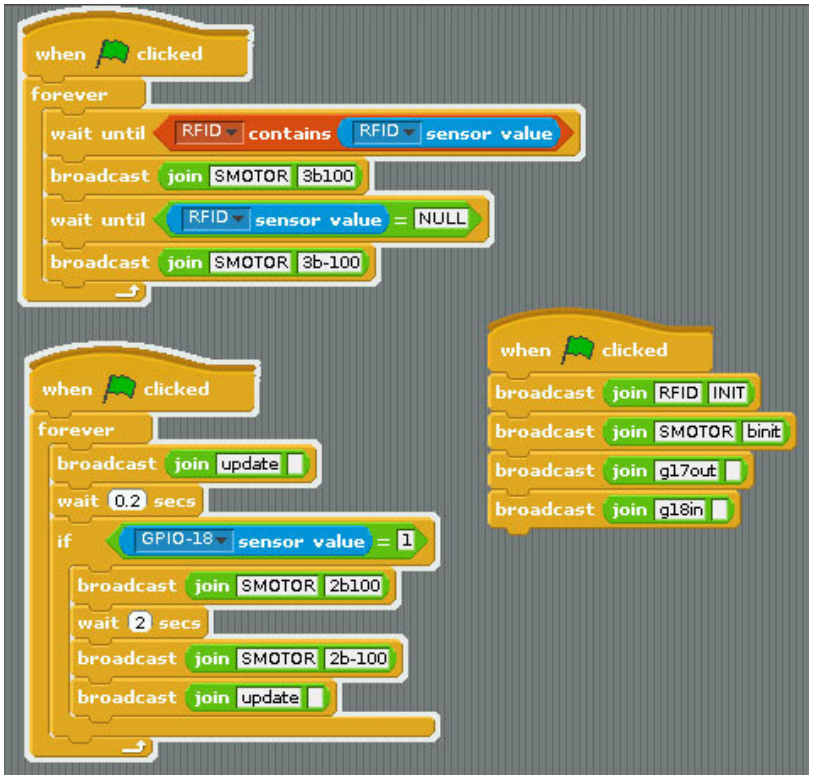

Scratch demo read RFID and GPIO output



RFID reader & Relay & Step Motor control

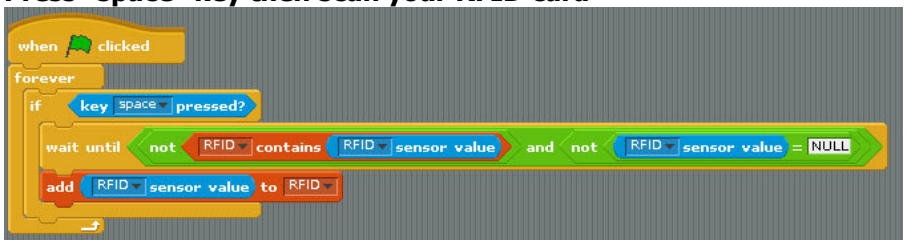
scratch demo code





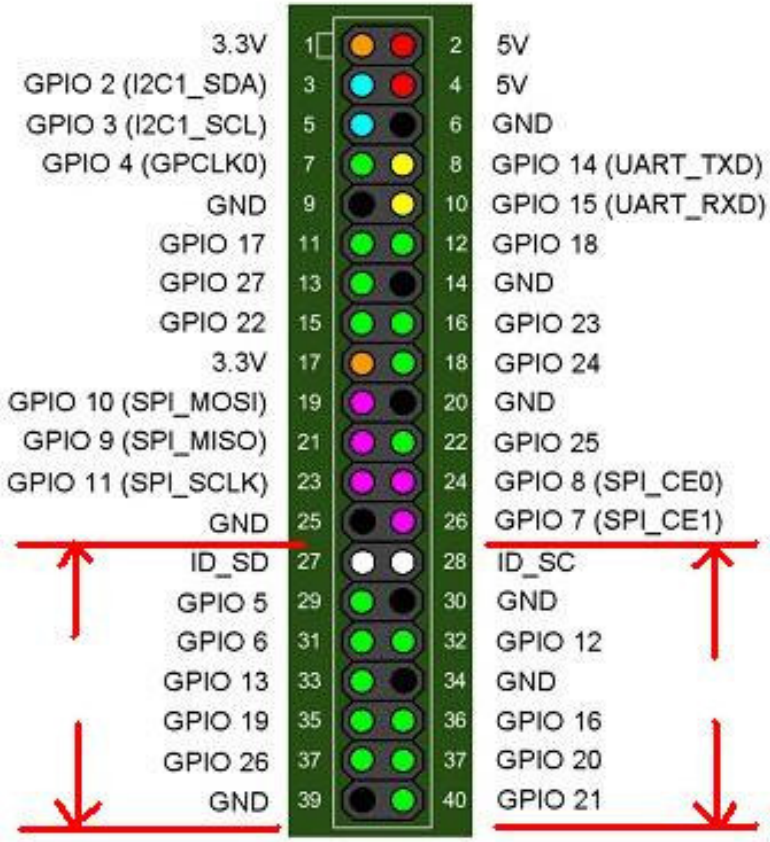
ADD RFID Card into data base

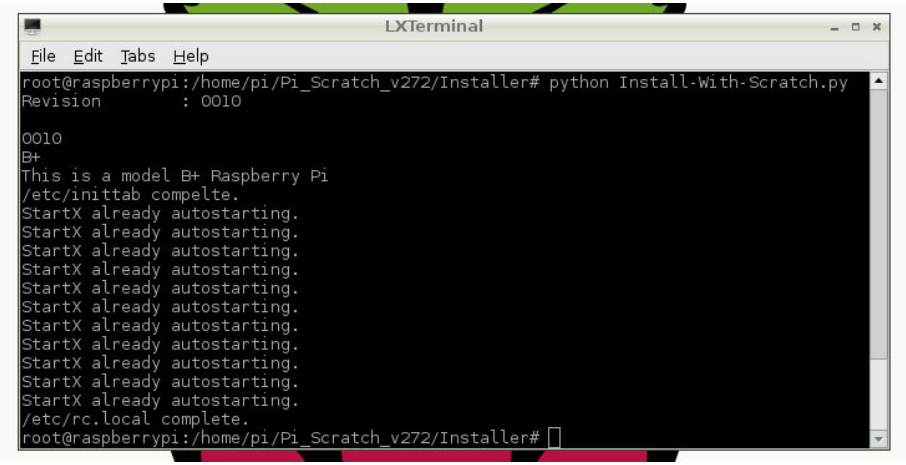
Press "space" key then scan your RFID card



Active with Step Motor and GPIO Switch input

31. Raspberry Pi B+ (B Plus) GPIO support

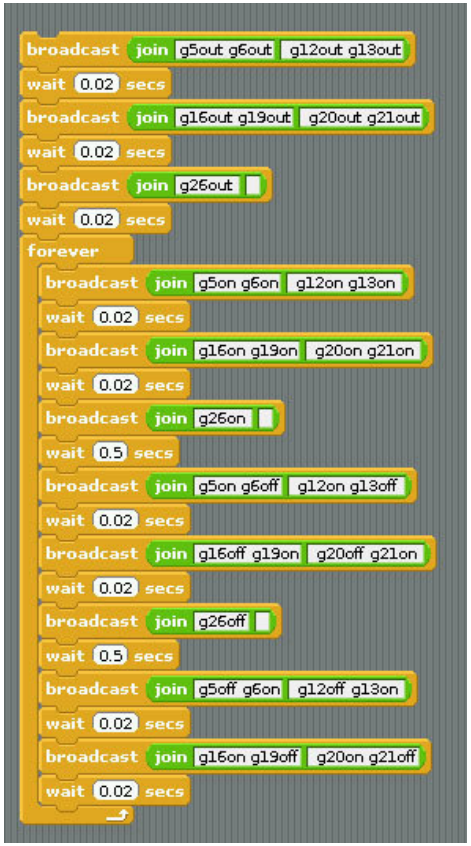




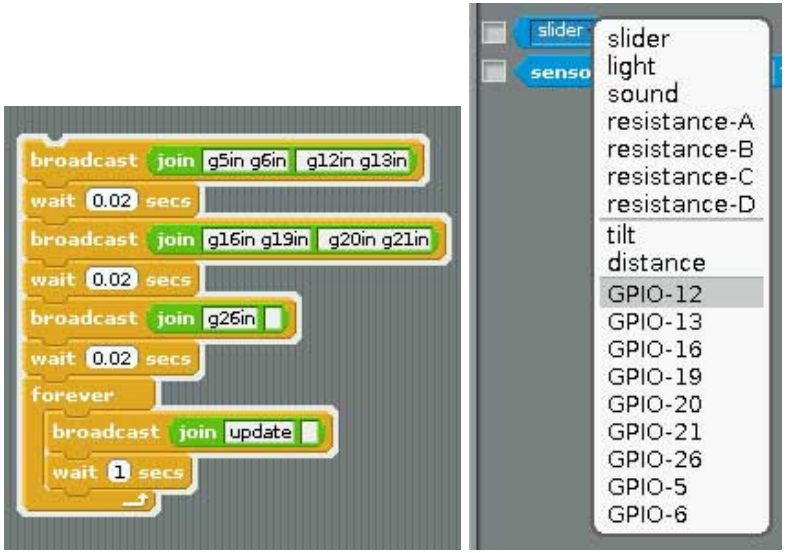
In Installer folder , type "python Install-With-Scratch.py" System will auto detect model B+, and install Pi_Scratch for model B+
In "Pi_Scratch_v272-B" folder

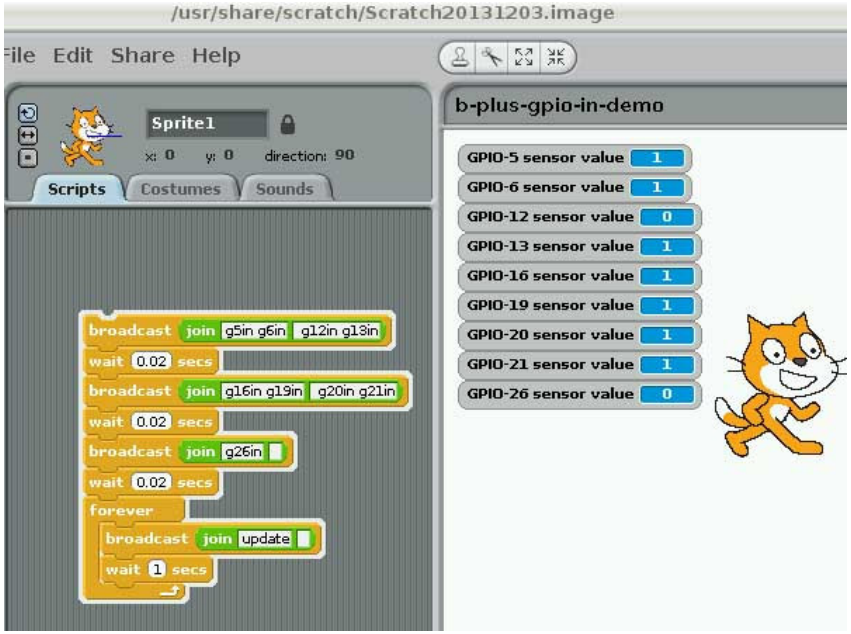
Example 01- Setting GPIO as output

1. define GPIO 5,6,12,13,16,19,20,21,26 as output
2. send GPIO 5,6,12,13,16,19,20,21,26 LED "ON"
3. send GPIO 6,13,16,20,26 LED "OFF" & 5,12,19,21 LED "ON"
4. send GPIO 6,13,16,20 LED "ON" & 5,12,19,21 LED "OFF"



Example 02- Setting GPIO as input



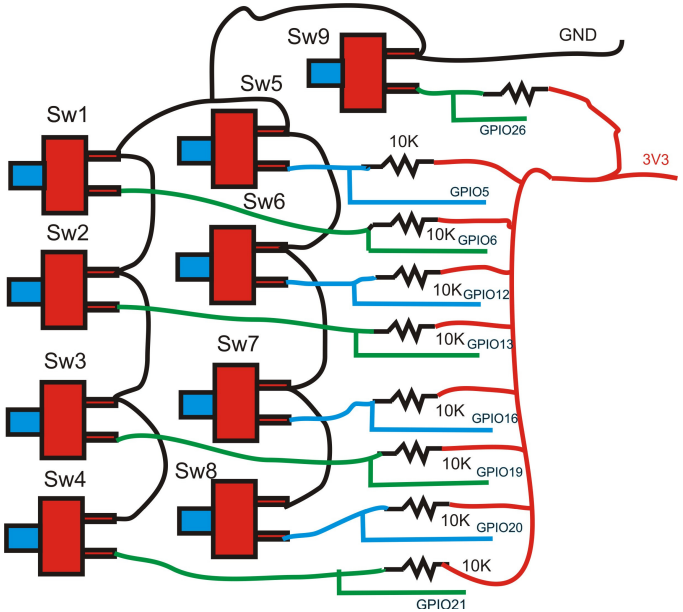


(1)"g5in g6in g12in g13in" setting GPIO 5, 6,12,13 as input pin
"g16in g19in g20in g21in" setting GPIO 16,19,20,21 as input pin
"g26in " setting GPIO 26 as input pin

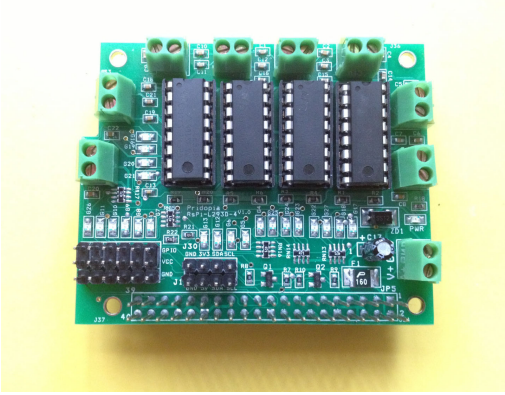
(2) broadcast "Update"

(3) in Sensing --> Slider , you will see the

" GPIO-12, GPIO-13, GPIO-16,GPIO-19, GPIO-20,GPIO-21, GPIO-26, GPIO-5,GPIO-6 " 9 GPIO in the list



32. Raspberry Pi B+ (B Plus) 8 DC Motor PWM & GPIO support

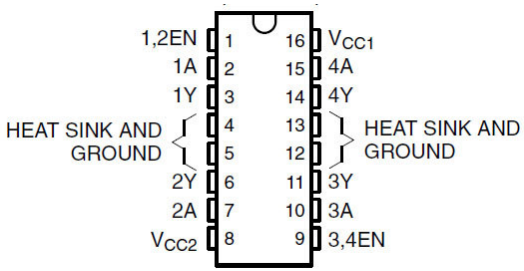


This board use RS-Pi
GPIO 17,18,27,22 for MA1, MB1
GPIO 23, 24, 25, 4 for MA2, MB2
GPIO 5, 6,12,13 for MA3,MB3
GPIO 16,19,20,21 for MA4,MB4

(1)Scratch PWM command mode

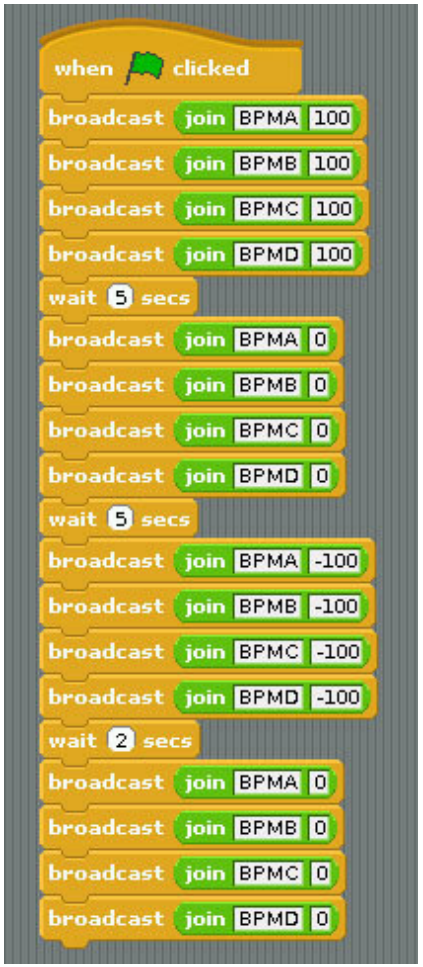
A - GPIO 16,19 Motor A B - GPIO 20,21 Motor B
C - GPIO 5,6 Motor C D - GPIO 12,13 Motor D

command "BPM"+"Motor Name"+ "speed"
speed (0) stop
speed (10 ~100) clockwise
speed (-10 ~ -100) anticlockwise

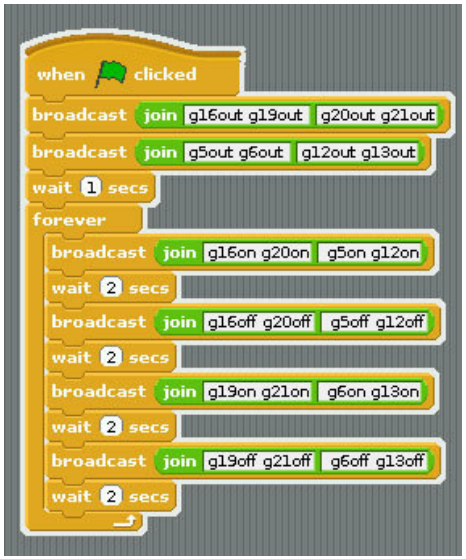


L293D

Pin 1	Pin 2	Pin 7	Function
High	Low	High	Turn clockwise
High	High	Low	Turn anti-clockwise
High	Low	Low	Stop
High	High	High	Stop
Low	Not applicable	Not applicable	Stop

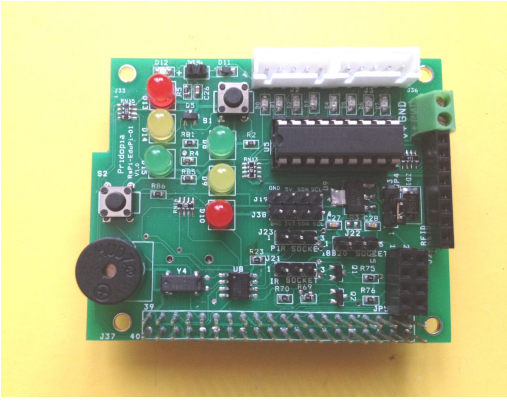


PWM mode command



GPIO Mode command

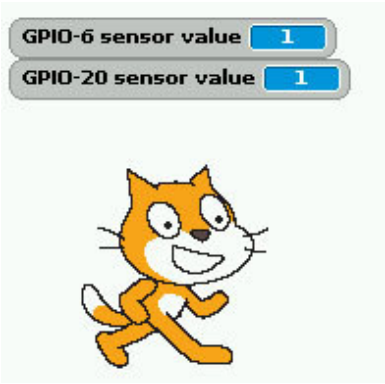
33. Raspberry Pi B+ (B Plus) EDU Learning Board support

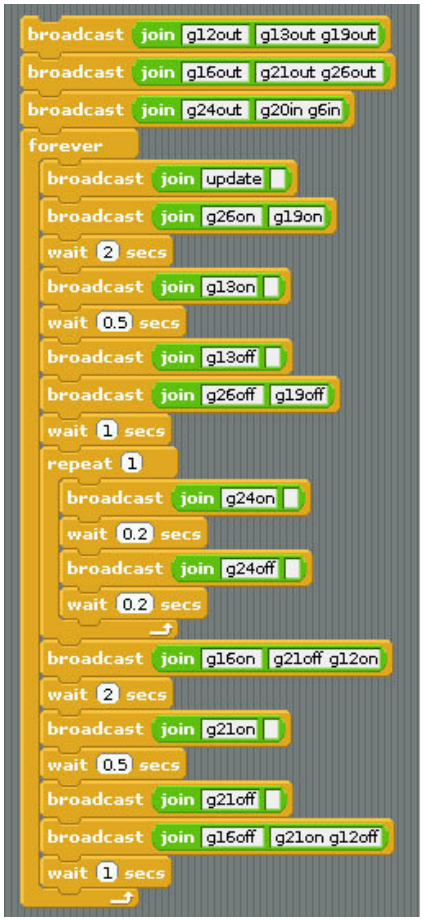


first set traffic light Red (GPIO19), Yellow(GPIO13), Green (GPIO12) ,Switch 1(GPIO20)

second set traffic light Red (GPIO16), Yellow(GPIO21), Green (GPIO26) ,Switch 2(GPIO6)

1. **Setting GPIO 19,13,12 as output for R,Y,G LED**
2. **Setting GPIO 16,21,26 as output for R,Y,G LED**
3. **Setting GPIO 20,6 as input for 2 Switch**

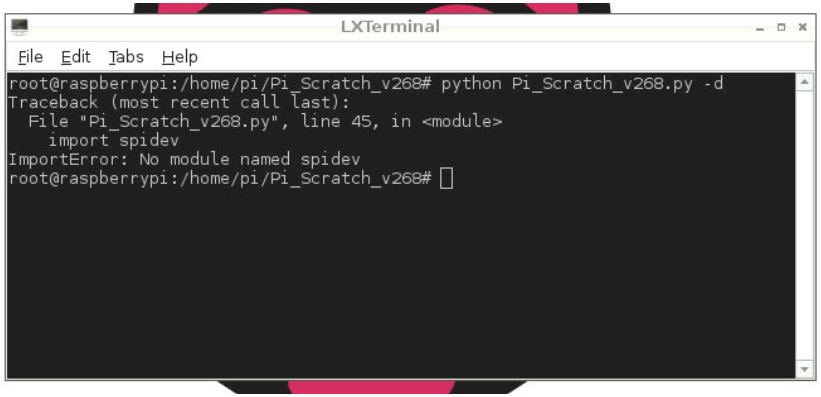




BASIC traffic light demo

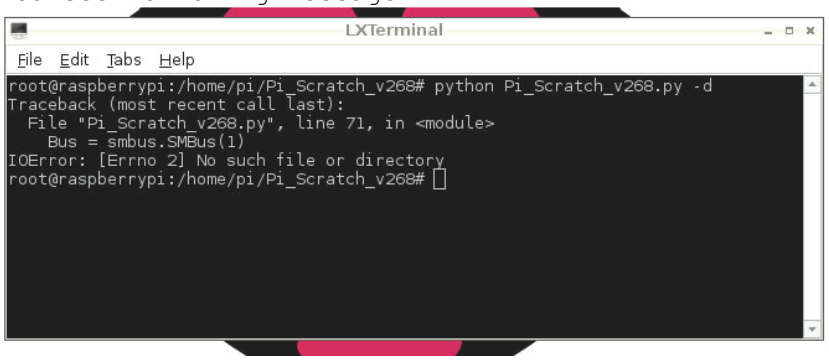
Pi _Scratch Install Q & A

1. When you type “python Pi_Scratch_v268.py -d”
You see following message

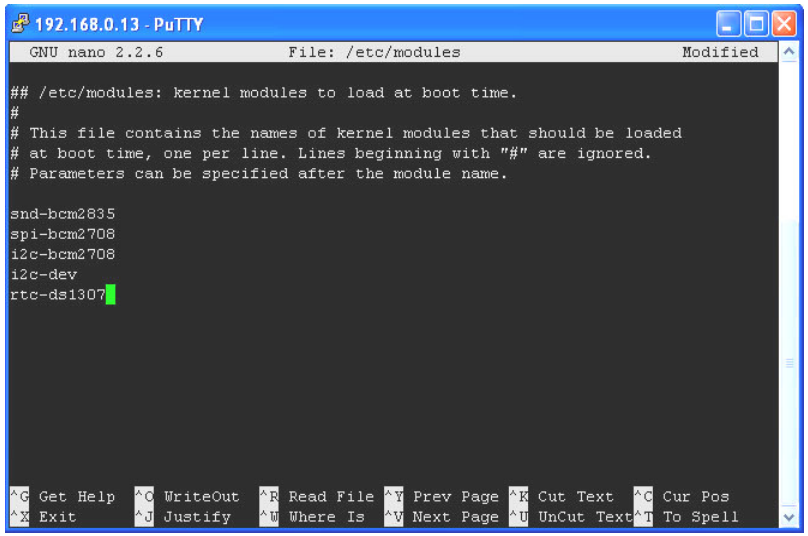


Type “python Install.py” install again

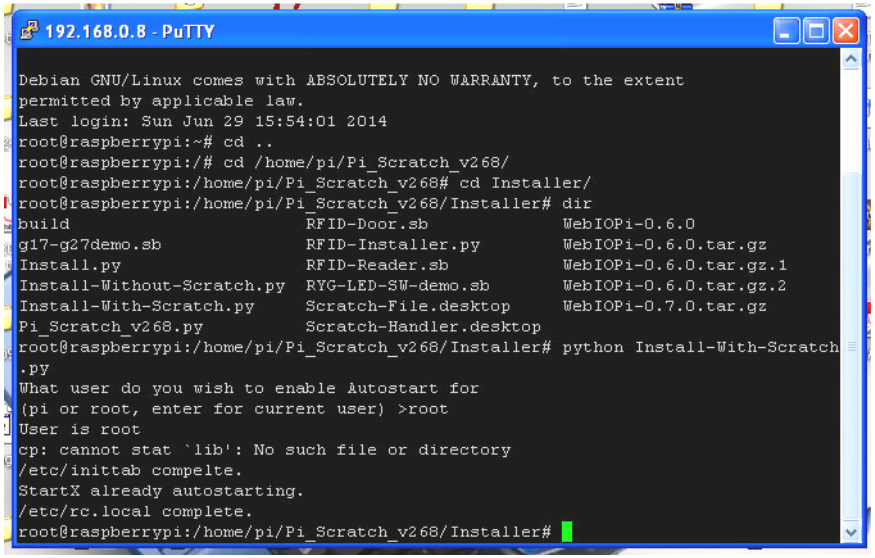
2. When you type “python Pi_Scratch_v268.py -d”
You see following message



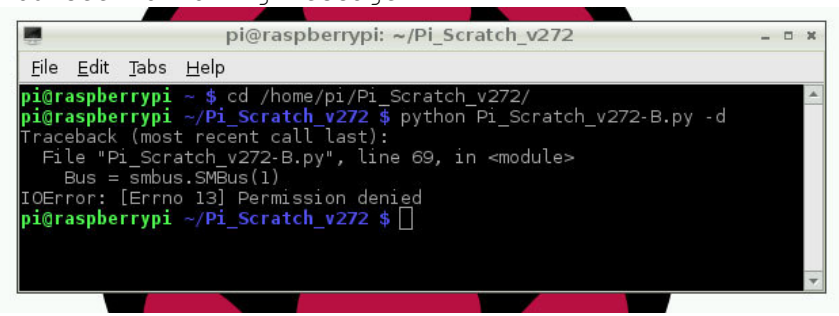
Check your setting in /etc/modules by use
sudo nano /etc/modules



3. Use manually start Pi_Scratch first, after everything working fine, then setting Auto login and auto start , use root first

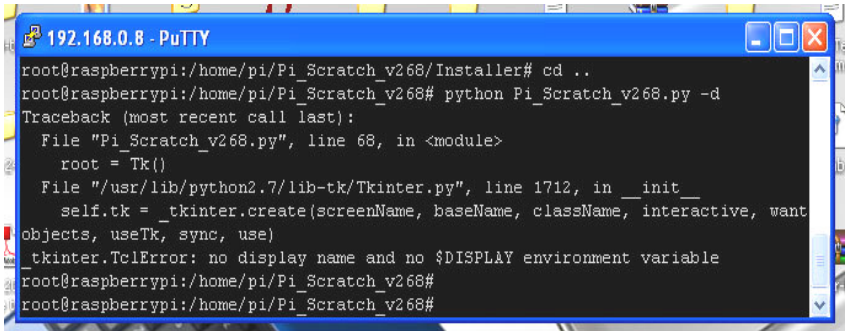


4. When you type “python Pi_Scratch_v272-B.py -d”
You see following message

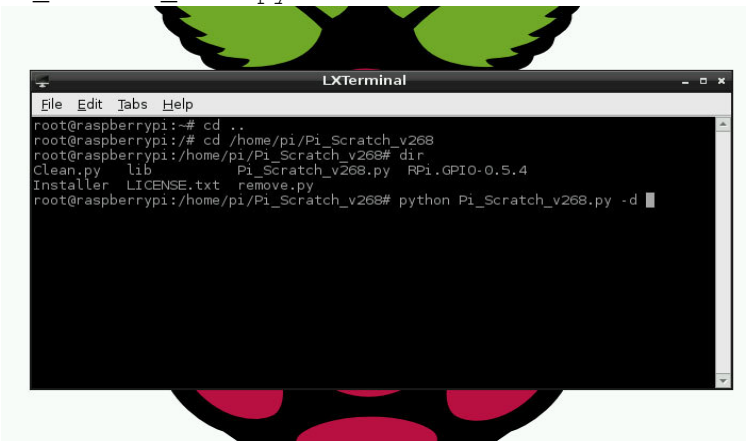


Use “root” to login your Pi, because some of the software function need access your Pi

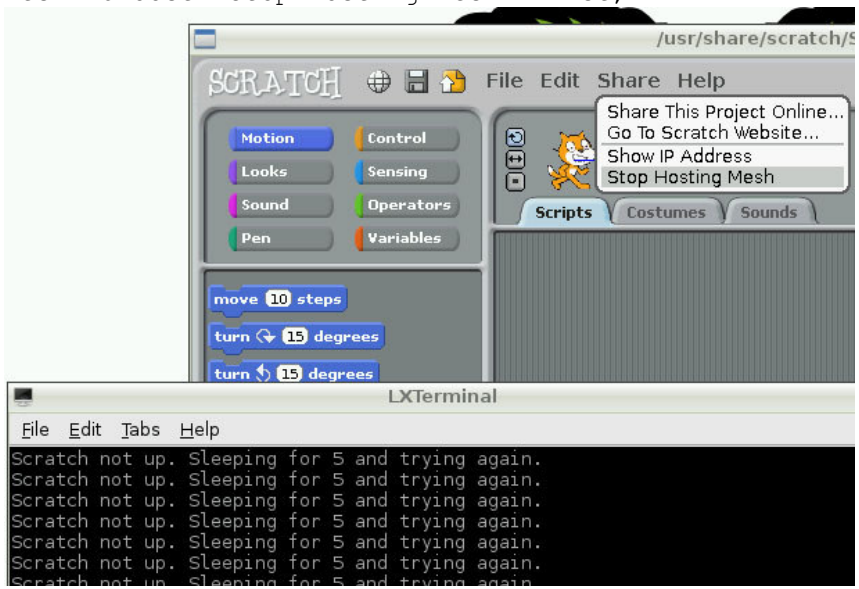
5. When you type “python Pi_Scratch_v268.py -d”
You see following message



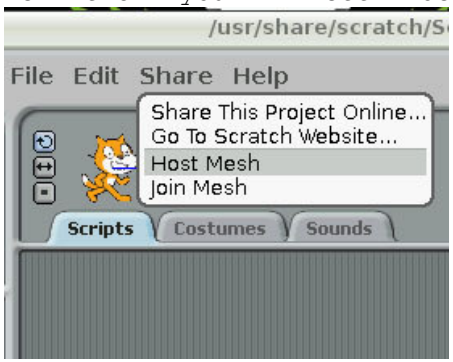
Type “startx” first, the open “LXTerminal” type “python Pi_Scratch_v268.py -d”



6. When Manually start Pi_Scratch if you can’t see “Host Mesh” Choose “Stop Hosting Mesh” first,



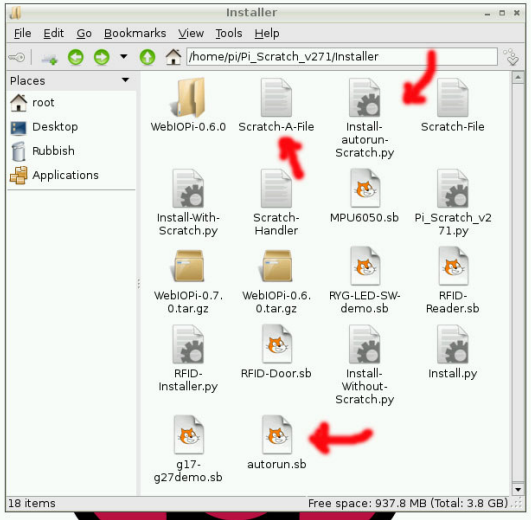
“shift S” you will see “Host Mesh”



How to auto run Pi_Scratch without press any key

Type “python Install-autorun-Scratch.py”

Will do the setting for you also copy the autorun.sb example file



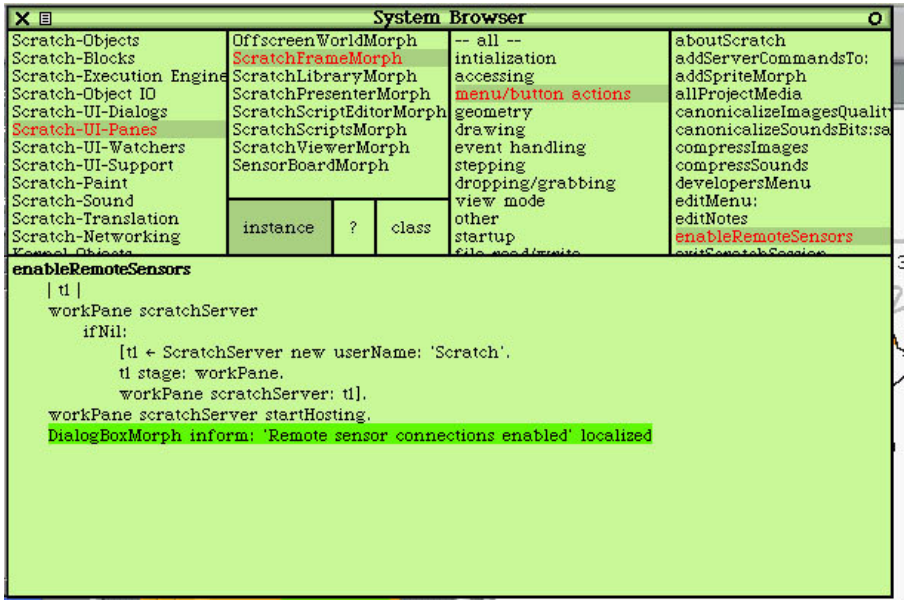
Use Pi_Scratch_V271

When receive “**startup**” to active auto run example file

We now have the program auto run in presentation mode, but as mentioned earlier there is a pop-up message **"Remote sensor connections enabled"** that stop update information for sensor “ it will update until you have clicked OK. This is something that needs to be turned off within the Scratch application, which is one of the hidden settings within Scratch.

After lauching Scratch shift left-click on the R in the Scratch logo (top left) and choose "turn fill screen off". Click on the white area to the right or bottom of the screen and choose "Open" and "Browser". Navigate to the section Scratch-UI-Panes -> ScratchFrameMorph -> menu/button actions -> enableRemoteSensors and remove the line:

Dialogboxmorph Inform: 'Remote Sensor Connections Enabled' Localized

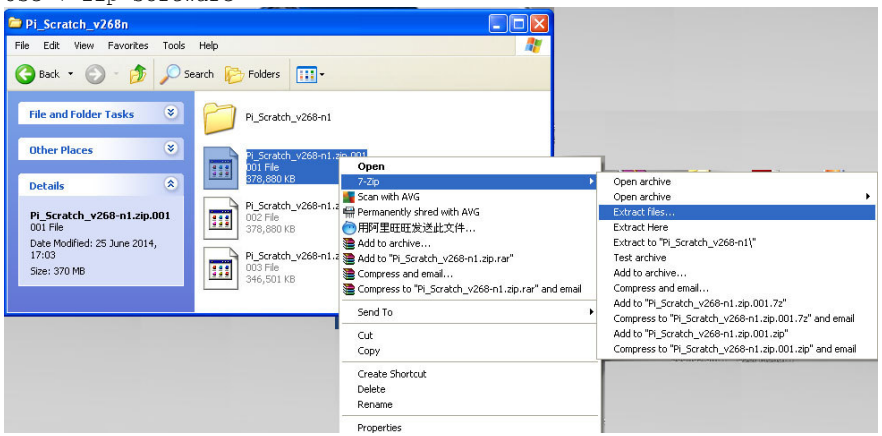


Then click with the centre mouse button (or Ctrl and left button) and choose Accept (s).

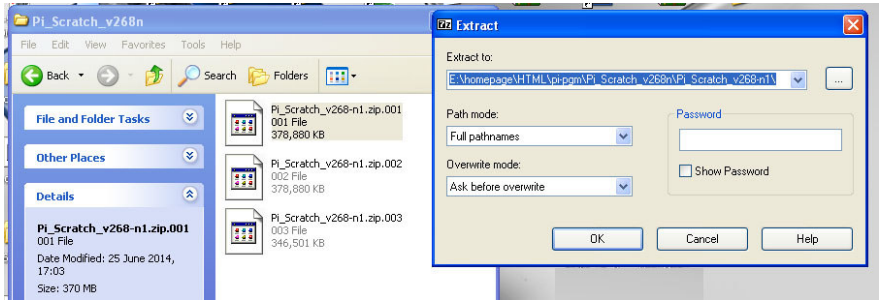
You can now remove the white space in Scratch by shift left click on the R in Scratch logo and choose Turn Fill screen on. Finally shift left click on the logo again and choose **"save image for end-user"**

How to use three zip file change to Image file

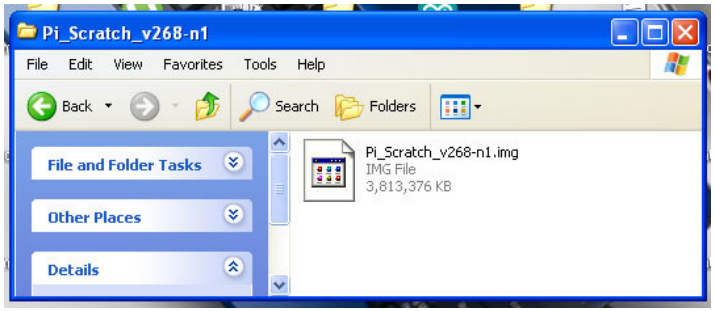
Use 7-Zip software



Copy three zip file into one new folder, use 7-zip, click right mouse key choose “Extract files”

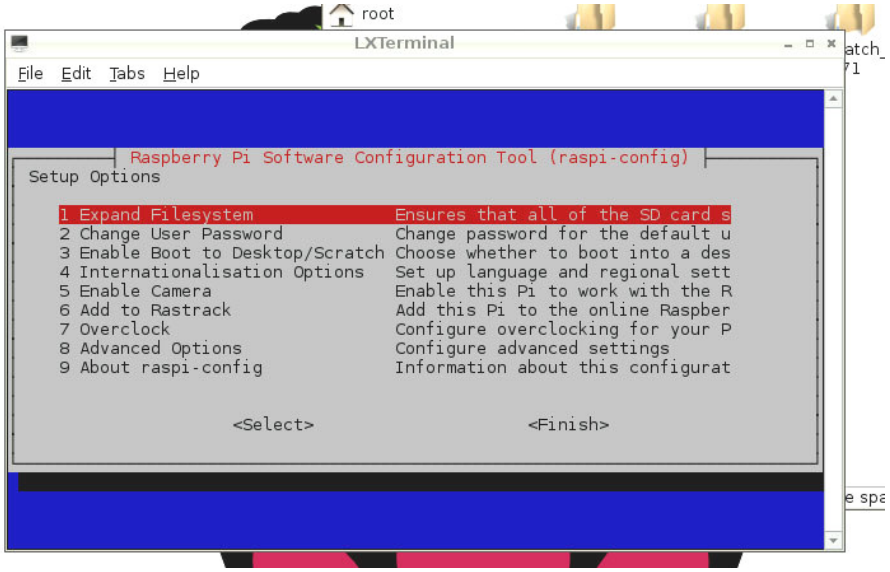


Click “OK”



After finished, you will have Pi_Scratch-V268-n1.img

First thing need do for Pi_Scratch ready Image



Use the Pi_Scratch ready Image first thing you need do

sudo raspi-config

Choose 1 “expand Filesystem to Ensures that all of the SD card s

How to upgrade from old Pi_Scratch to new Ver Pi_Scratch

In old ver Pi_Scratch_268 folder --- remove auto-login auto run first

sudo python remove.py

then reboot

download new Ver Pi_Scratch

sudo wget http://pridopia.co.uk/pi-pgm/Pi_Scratch_v272.tar

tar xf Pi_Scratch_v272.tar
cd Pi_Scratch_v272
cd Installer
sudo python Install-autorun-Scratch.py --- auto run example without press any key - choose root

or

sudo python Install-with-Scratch.py --- auto load example but need click "ok" and "greeng flag" in scratch - choose root

Pi_Scratch-v272 ready Image information

Use Raspbian Wheezy 2014-06-20

ready to use Pi_Scratch_v272 with Auto Login & Auto Load Pi_Scratch & SSH enable --

SSH enable -- user - root password : pi for 4GB SD card

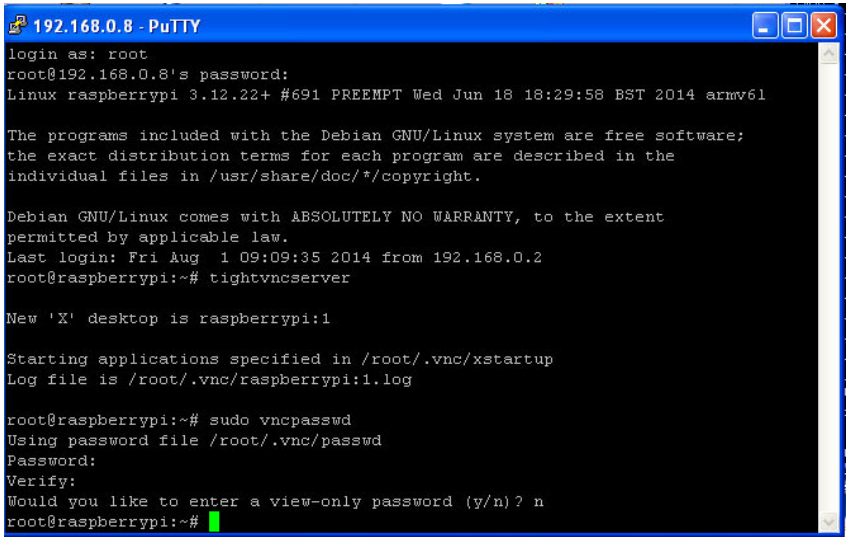
TightVNCserver install password : pipipi --

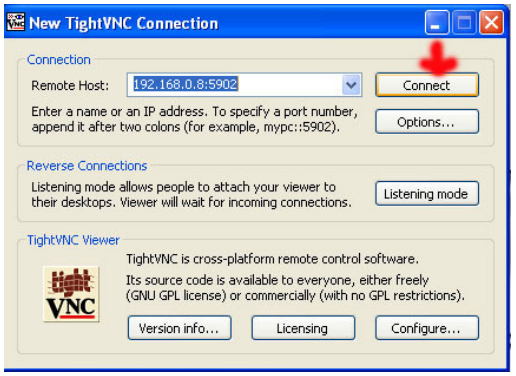
Use **sudo vncpasswd** to change password

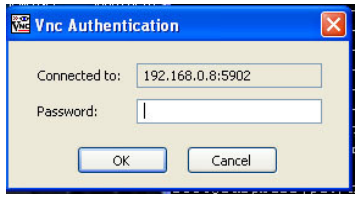
VNC (VIRTUAL NETWORK COMPUTING)

VNC is a graphical desktop sharing system that allows you to remotely control the desktop interface of one computer from another. It transmits the keyboard and mouse events from the controller, and receives updates to the screen over the network from the remote host. You will see the desktop of the Raspberry Pi inside a window on your computer. You'll be able to control it as though you were working on the Raspberry Pi itself.

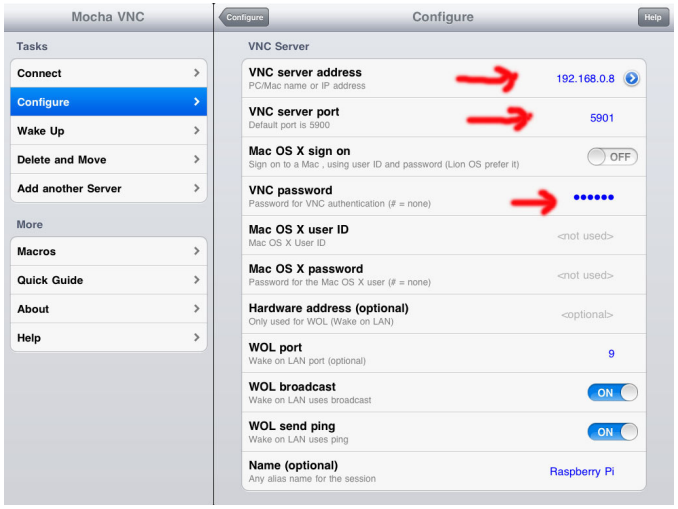
Type **tightvncserver** to start VNC control from PC Mac or iPad

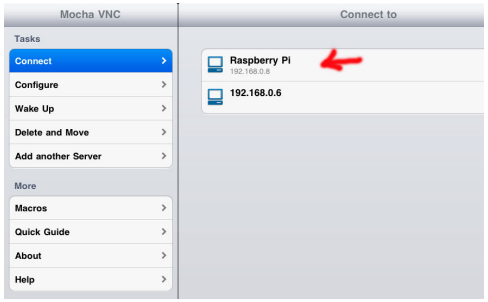


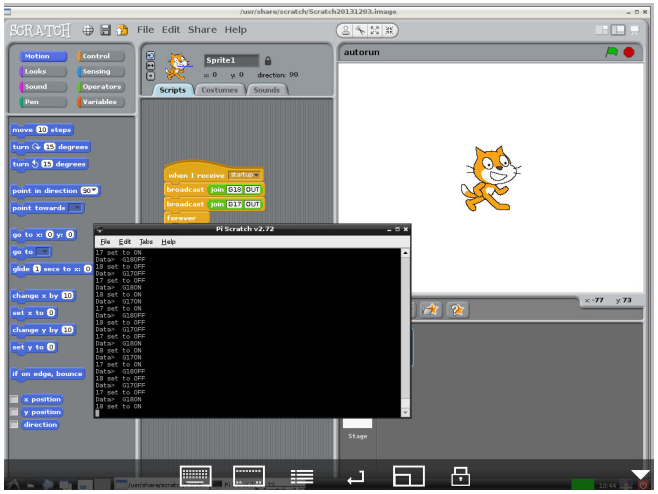




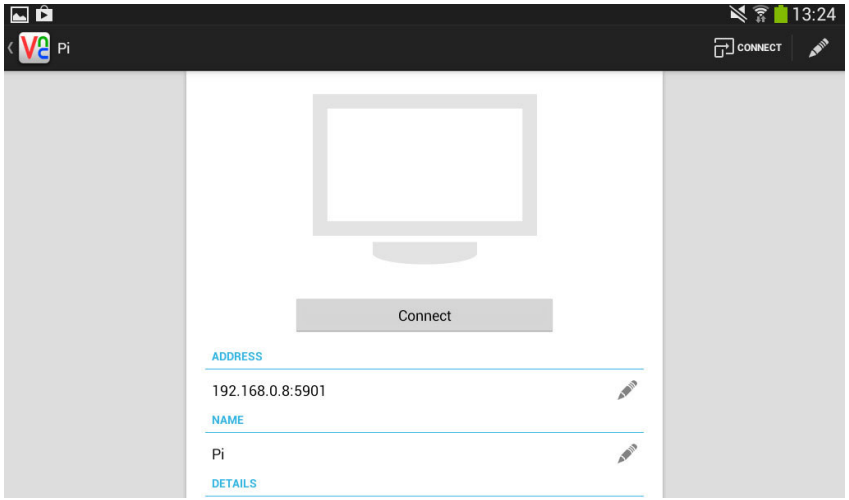
VNC connect from iPad & idevice

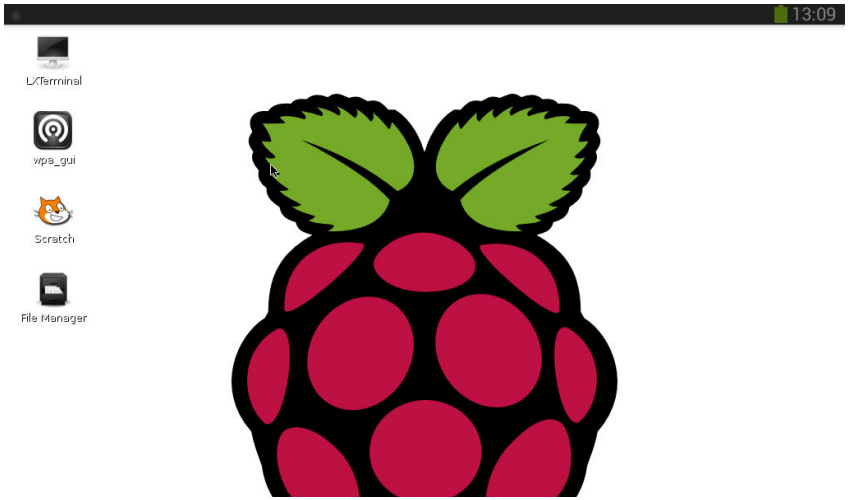






VNC connect from Android Smart Phone / Tablet

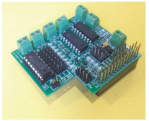




http://www.pridopia.co.uk/ixx-rspi.html

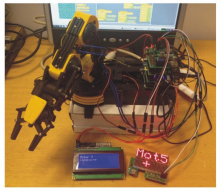
Find all your needs for Raspberry Pi in one place - Pridopia

DC Motor Series

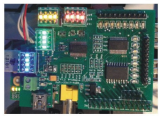


Rs-Pi-L293D-3
6 Motor Board

Rs-Pi-L293D-1
2 Motor Board



Rs-Pi-L293D-3 6 Motor-
i2c 20x4 LCD &
24 x16 LED Matrix Demo



PWM LED set demo
(Red, Green, Yellow, Blue)

RFID Reader

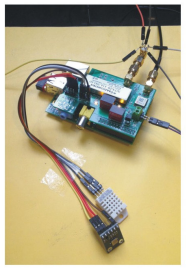
Rs-Pi-RFID Reader
&Relay & Step Motor
control

Rs-Pi-4 Relay
& Step Motor Board

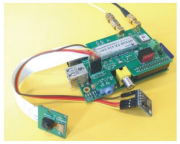
LED Matrix

Rs-Pi- LED 24x16
Matrix Board

Rs-Pi- LED 16x16
Matrix Board

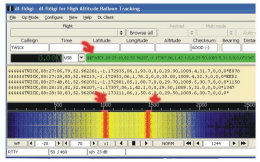


Rs-Pi-P-HAB 5V Board
(High-Altitude Balloon)



Rs-Pi-P-HAB 3.3V Board
(High-Altitude Balloon)

Rs-Pi-P-HAB & GPS Antenna
FM Antenna &
USB SDR Encoder



Pridopia Limited
New Brook House
385 Alfreton Road
Nottingham NG7 5LR
United Kingdom
TEL: 0844-2492899

Software for Scratch
http://www.pridopia.co.uk/rs-pi-set-scratch.html

Make Raspberry Pi More Easy !

http://www.pridopia.co.uk/ixx-rspi.html

Find all your needs for Raspberry Pi in one place - Pridopia

4 USB Hub Series

Rs-Pi-4 USB Hub +
Multi-function i2c Board

Rs-Pi-4 USB Hub +
32 GPIO(i2c/SPI) Board

7 USB Hub Series

Rs-Pi-7 USB Hub +
Multi-function i2c Board

Rs-Pi-7 USB Hub +
16 GPIO (i2c/SPI) Board

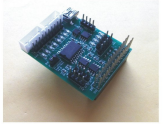
PWM / Servo Series

Rs-Pi-32 channel
PWM/Servo Board

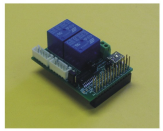
Rs-Pi-16 channel PWM
& 16 GPIO (i2c/SPI)Board

Rs-Pi-16 channel PWM
& 4 Motor - PSU Board

Relay & Motor Series

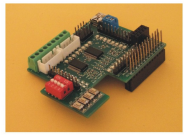


Rs-Pi- uln2803
SPI AD/DA -1-Wire
Multi-function Board

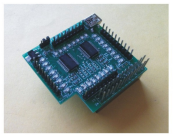


Rs-Pi- 2 Relay +
Step Motor Board

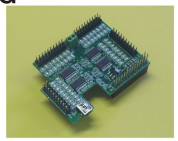
i2c & SPI GPIO Board



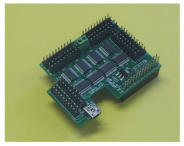
Rs-Pi-Multi-function
32 GPIO (i2c/SPI) Board



Rs-Pi-32 GPIO
(i2c/SPI) Board



Rs-Pi-64 GPIO
(i2c/SPI) Board

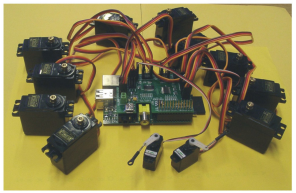


Rs-Pi-128 GPIO
(i2c/SPI) Board

Rs-Pi- LED 24x16
Matrix Board

Rs-Pi-LCD-Master
Board (84x48,16x2,20x4)

Rs-Pi- RS232
console cable



Pridopia Limited
New Brook House
385 Alfreton Road
Nottingham NG7 5LR
United Kingdom
TEL: 0844-2492899

Software for Scratch
http://www.pridopia.co.uk/rs-pi-set-scratch.html

Make Raspberry Pi More Easy !