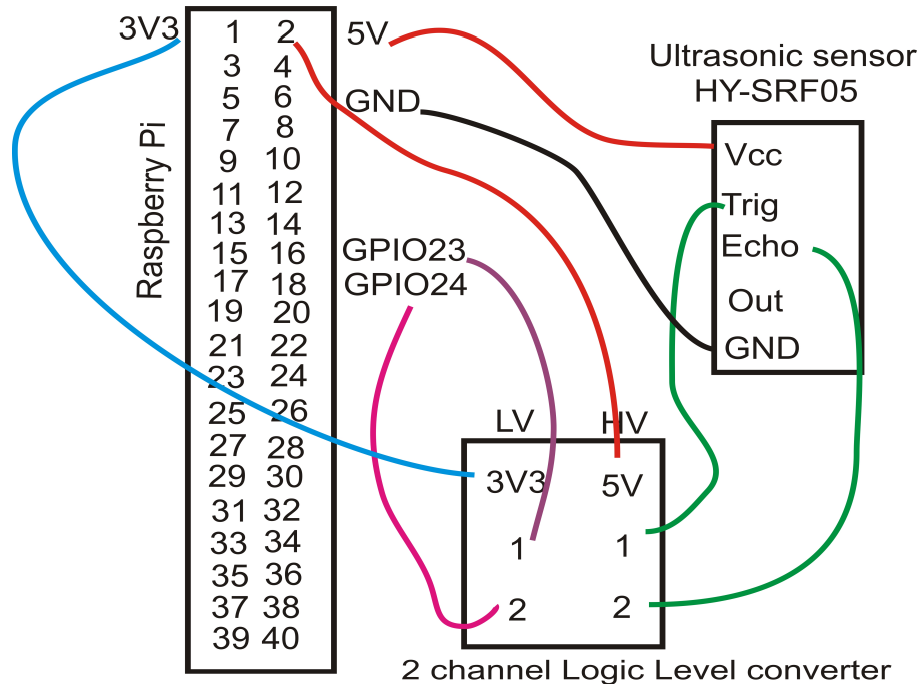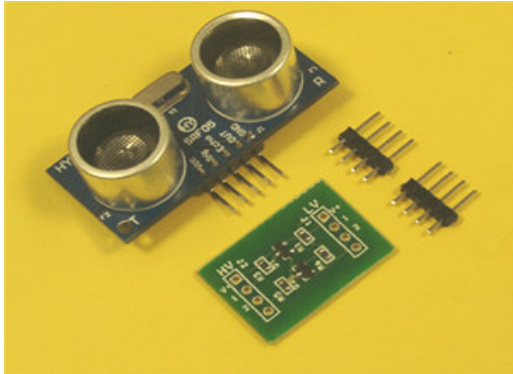# Ultrasonic   distance sensor (HY-SRF05)
with 2channel Logic Level converter

Use two GPIO Trig(T)   Echo(E)
Update --> check distance and update
          T23   -->   Trig use GPIO23
          E24   -->   Echo use GPIO24

Ultrasonic sensor module Description :

- Working Voltage : 5V(DC)
- Static current: Less than 2mA.
- Output signal: Electric frequency signal, high level 5V, low level 0V.
- Sensor angle: Not more than 15 degrees.
- Detection distance: 2cm-450cm.
- High precision: Up to 0.2cm
- Input trigger signal: 10us TTL impulse
- Echo signal : output TTL PWL signal

- Mode of connection:
- 1.VCC   2.trig(T)   3.echo(R)   4.OUT   5.GND

- Use method:
- Supply module with 5V, the output will be 5V while obstacle in range, or 0V if not. The out pin of this module is used as a switching output when anti-theft module, and without the feet when ranging modules.

2 channel Logic Level Converter Description :

The logic level converter is a small PCB which safely allows you to interface devices which steps down 5V signals to 3.3V signals and step up 3.3V signals to 5V.

The converter has 2, inputs and outputs and can be used with I2C, SPI, UART, etc.

The board needs to be powered via two sources, 5V and 3.3V and your data lines are linked via pins 1 to 2 on each side so an input on pin 1 will be converted to the lower or higher output on the adjoining pin 1
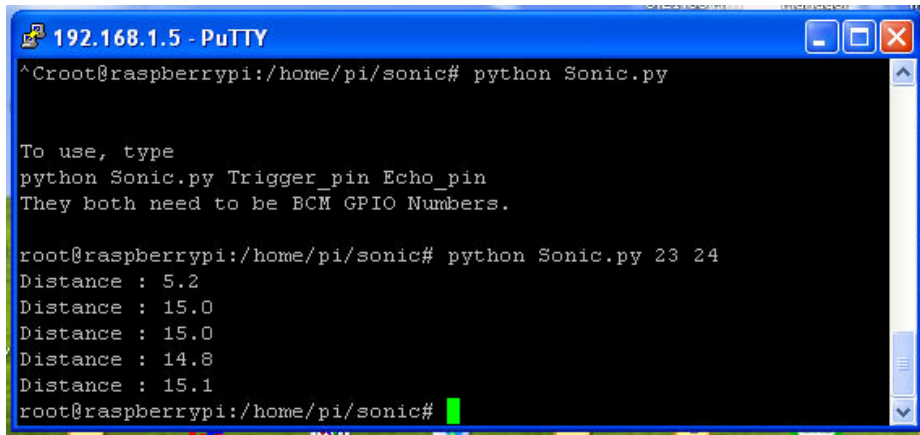
Download GPIO library

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

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

Raspberry Pi

3V3	1	2	5V
3	4
5	6	GND
7	8
9	10
11	12
13	14
GPIO23	15	16
GPIO24	17	18
19	20
21	22
23	24
25	26
27	28
29	30
31	32
33	34
35	36
37	38
39	40

Ultrasonic sensor
HY-SRF05

Vcc
Trig
Echo
Out
GND

LV   HV
3V3   5V
1   1
2   2

2 channel Logic Level converter

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

```
192.168.1.5 - PuTTY
^Croot@raspberrypi:/home/pi/sonic# python Sonic.py


To use, type
python Sonic.py Trigger_pin Echo_pin
They both need to be BCM GPIO Numbers.

root@raspberrypi:/home/pi/sonic# python Sonic.py 23 24
Distance : 5.2
Distance : 15.0
Distance : 15.0
Distance : 14.8
Distance : 15.1
root@raspberrypi:/home/pi/sonic#
```

Sonic.py demo

command   sudo python Sonic.py 23 24
 23 Trigger pin   24 Echo pin (you can use other GPIO pin)

Demo program download from our web site
http://www.pridopia.co.uk/pi-sonic-level.html