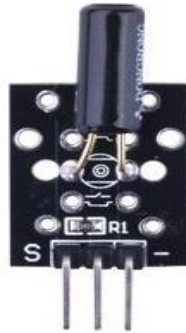# Shock Switch

Overview

A shock switch (also called a shock sensor or vibration switch) senses shock or vibration and translates it into a signal that can switch a circuit on or off. (Unlike a knock switch, it detects changes in position rather than physical impacts.) In this experiment you'll make your Raspberry Pi turn on an LED light whenever a shock switch detects vibrations.

Experimental Materials

```
Raspberry Pi     x1
Breadboard       x1
Shock Switch     x1
LED (3 pin)      x1
Resistor (330Ω)  x1
Dupont jumper wires
```

Experimental Procedure

1. If you have not done so already, prepare your development system by installing the Python interpreter, RPi.GPIO library, and wiringPi library as described in READ_ME_FIRST.TXT.

2. Install the shock switch and three-pin LED on your breadboard, and use the resistor and Dupont jumper wires as illustrated in the Wiring Diagram below. Note you will connect only two of the three pins on the LED.

3. Execute the sample stored in this experiment's subfolder.
   If using C, compile and execute the C code:

```
cd Code/C
gcc shockSwitch.c -o shockSwitch.out –lwiringPi
shockSwitch.out
```

If using Python, launch the Python script:

```
cd Code/Python
python shockSwitch.py
```

4. Make experimental observations. As you shake the vibration switch (or your entire breadboard), the LED illuminates. When the vibration switch returns to level, the LED turns off. You may also discover the switch can act as a tilt sensor, and consistently illuminate when tilted horizontally beneath the level in one direction, while remaining off when tilted in the opposite direction.

Wiring Diagram
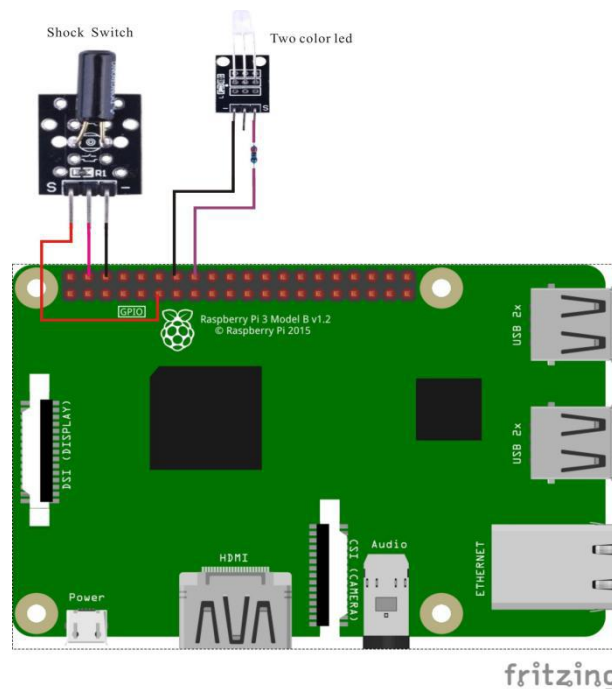
Shock Switch pin position:

S	↔	Raspberry Pi Pin 11
"+"	↔	Raspberry Pi +5V
"-"	↔	Raspberry Pi GND

LED pin position:

"S"	↔	Raspberry Pi 16 (through resistor)
"-"	↔	Raspberry Pi GND

Sample Code

Python Code

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

ShockPin = 11
LedPin   = 16

Led_status = 0

def setup():
    GPIO.setmode(GPIO.BOARD)   # Numbers GPIOs by physical location
    GPIO.setup(LedPin, GPIO.OUT)   # Set LedPin's mode is output
    GPIO.setup(ShockPin, GPIO.IN,  pull_up_down=GPIO.PUD_UP)


def swLed(ev=None):
    global Led_status
    Led_status = not Led_status
    GPIO.output(LedPin, Led_status)  # switch led status(on-->off;
off-->on)
    print "led: " + ("on" if Led_status else "off")

def loop():
    GPIO.add_event_detect(ShockPin, GPIO.FALLING, callback=swLed,
bouncetime=200) # wait for falling
    while True:
        pass   # Don't do anything

def destroy():
    GPIO.output(LedPin, GPIO.LOW)     # led off
    GPIO.cleanup()                    # Release resource

if __name__ == '__main__':    # Program start from here
    setup()
    try:
        loop()
    except KeyboardInterrupt:  # When 'Ctrl+C' is pressed, the child
program destroy() will be  executed.
        destroy()
```

C Code

```c
#include <wiringPi.h>
#include <stdio.h>

#define ShockPin     0
#define LedPin       4

void myISR(void)
{
    printf("shock occured.\n");
    if(digitalRead(LedPin) == HIGH)
    {
        digitalWrite(LedPin, LOW);
    }
    else
    {
        digitalWrite(LedPin, HIGH);
    }
}


int main(void)
{
    if(wiringPiSetup() == -1)
    {
        printf("setup wiringPi failed !\n");
        return 1;
    }

    pinMode(ShockPin,  INPUT);
    pinMode(LedPin,    OUTPUT);

    if(wiringPiISR(ShockPin, INT_EDGE_FALLING, &myISR) == -1)
    {
        printf("setup ISR failed !");
        return 1;
    }
    while(1);

    return 0;
}
```

Technical Background

Shock switches are widely used in toys, burglar alarms and other applications. This module is based on the SW-520D sensor, which contains a ball whose motion—during vibration—triggers a metal switch. The electrical characteristics are similar to old-fashioned mercury switches, but without the environmental dangers associated with mercury's toxicity. When the switch is in the quiescent state, the switch conducts (is "on") if one terminal is below the level of 15° degrees; when the opposite terminal is 15° below horizontal, the switch is off. The switch is only suitable for triggering a small current (5mA) circuit—such as an LED or transistor—and is not suitable for use as a power switch. The module is soldered with a 10KΩ pull-up resistor connected to the +5V input.