Mini Reed Switch

Overview

A reed switch is a type of switch in which the open gap between two wires separated in a sealed glass tube can be closed by introducing the presence of a nearby magnet. Compared to more recently developed Hall effect sensors (which also detect magnetic fields), they are electro-mechanical rather than solid-state in operation. However, over a long history of improvement, reed switches' reliability and low cost have kept them popular in many applications, such as airbag mechanisms in automotive safety systems.

This experiment uses the Raspberry Pi to drive an LED that illuminates when a magnet is positioned near the reed switch.

Experimental Materials

Raspberry Pi	x1
Breadboard	x1
Mini Reed Switch sensor	x1
LED (3 pin)	x1
Resistor (330Ω)	x1
Dupont jumper wires	
Any magnet	(you provide)

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 mini reed switch sensor, three-pin LED and resistor on your breadboard, and use Dupont jumper wires to connect them to each other and your Raspberry Pi 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:

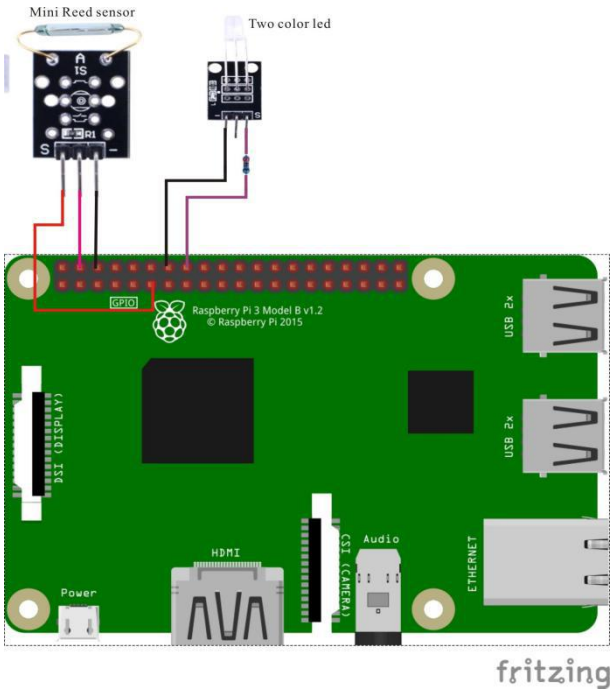
```
cd Code/C
gcc miniReedSwitch.c -o miniReedSwitch.out -lwiringPi
./ miniReedSwitch.out
```

   If using Python, launch the Python script:

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

4. Make experimental observations. When you hold your magnet close to the sensor, the LED comes on, and goes off when you remove the magnet.

Wiring Diagram



Mini Reed Switch pin position:

"S" ↔ Raspberry Pi pin 11

"+" ↔ Raspberry Pi +5V

"-" ↔ Raspberry Pi GND

LED pin position:

"S" ↔ Raspberry Pi pin 16(through resistor)

"-" ↔ Raspberry Pi GND

Sample Code

Python Code

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

ReedPin = 11
LedPin  = 16

def setup():
   GPIO.setmode(GPIO.BOARD)
   GPIO.setup(LedPin, GPIO.OUT)
   GPIO.setup(ReedPin, GPIO.IN, pull_up_down=GPIO.PUD_UP)
   GPIO.output(LedPin, GPIO.LOW)
def loop():
   while True:
      if(GPIO.input(ReedPin) == 0):
         print "Magnet detected - LED on!"
         GPIO.output(LedPin,GPIO.HIGH)
      else:
         print "No magnet detected - LED off!"
         GPIO.output(LedPin, GPIO.LOW)
       time.sleep(0.2)
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:
      destroy()
```

C Code

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

#define    reedPin    0
#define     LedPin    4

int cnt = 0;

int main(void)
{
   if(wiringPiSetup() == -1)
   {
      printf("setup wiringPi failed !\n");
      return -1;
   }
   pinMode(LedPin, OUTPUT);
   pinMode(reedPin, INPUT);
   pullUpDnControl(reedPin, PUD_UP);
   while(1)
   {
      if(!digitalRead(reedPin))
      {
         printf("Magnet detected...\n");
         digitalWrite(LedPin, HIGH);
      }
      else
      {
         printf("No magnet detected...\n");
         digitalWrite(LedPin, LOW);
      }
      delay(200);
   }
   return 0;
}
```