Hall Magnetic Sensor

Overview

We will use the Raspberry Pi to capture the signal of the Hall magnetic sensor and controls the LED's state based on the captured signal.

Materials Needed

RaspberryPi \*1

Breadboard \*1

Hall magnatic sensor \*1

ADC0832 \*1

Led \*1

Dupont Line

Preparation

1. Install python interpreter in your Raspberry Pi system

2. Install the RPi.GPIO library in your Raspberry Pi system

3. Install the wiringPi library in your Raspberry Pi system

See the attached <<Installing a Python Interpreter and Corresponding Libraries in a Raspberry Pi System>> for details.

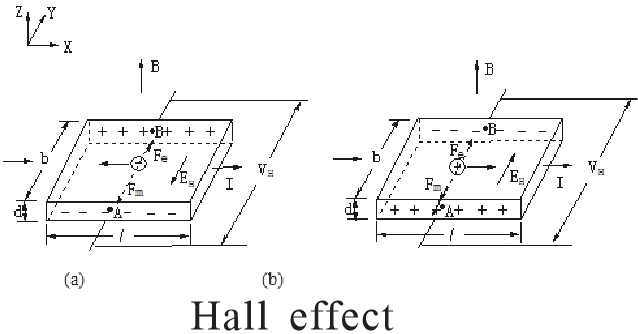
Description

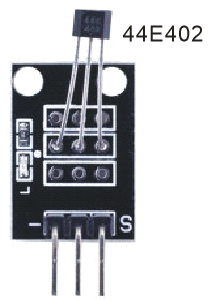
Brief Introduction

Hall Effect: When there is current at both ends of the semiconductor sheet and a uniform magnetic field with a magnetic induction strength B is applied in the vertical direction of the sheet, a Hall with a potential difference of UH will be generated in the direction perpendicular to the current and the magnetic field. Voltage.

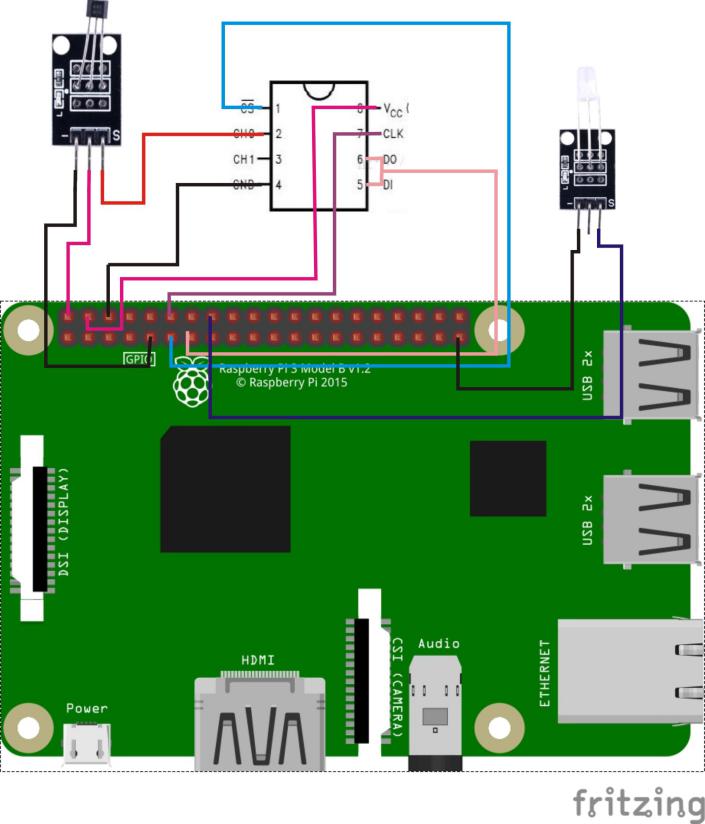
According to the Hall effect, a component made of a semiconductor material is called a Hall element. It has the advantages of being sensitive to magnetic field, simple structure, small volume, wide frequency response, large output voltage change, long service life, etc. Therefore, it has been widely used in measurement, automation, computer and information technology.

A voltage difference is generated when the Hall element and the magnet meet in the forward direction, and there is no voltage difference when the Hall element and the magnet meet in the forward direction, so that the voltage change can be obtained by the Raspberry Pi and the proximity of the magnet can be determined and use this signal to control on and off and the LED.





Cable Connection



Sample code

1. Python Code

#!/usr/bin/env python

import ADC0832

import time

import RPi.GPIO as GPIO

LedPin = 16

thresholdVal = 150

def init():

ADC0832.setup()

GPIO.setup(LedPin, GPIO.OUT)

def loop():

while True:

analogVal = ADC0832.getResult(0)

print 'analog value is %d' % analogVal

if(analogVal > thresholdVal):

GPIO.output(LedPin, GPIO.HIGH)

else:

GPIO.output(LedPin, GPIO.LOW)

time.sleep(0.2)

if \_\_name\_\_ == '\_\_main\_\_':

init()

try:

loop()

except KeyboardInterrupt:

ADC0832.destroy()

print 'The end !'

2. C Code

#include <wiringPi.h>

#include <stdio.h>

#include <string.h>

#include <errno.h>

#include <stdlib.h>

#define ADC\_CS 0

#define ADC\_CLK 1

#define ADC\_DIO 2

#define LedPin 4

#define thresholdVal 150

typedef unsigned char uchar;

typedef unsigned int uint;

uchar get\_ADC\_Result(void)

{

uchar i;

uchar dat1=0, dat2=0;

digitalWrite(ADC\_CS, 0);

digitalWrite(ADC\_CLK,0);

digitalWrite(ADC\_DIO,1); delayMicroseconds(2);

digitalWrite(ADC\_CLK,1); delayMicroseconds(2);

digitalWrite(ADC\_CLK,0);

digitalWrite(ADC\_DIO,1); delayMicroseconds(2);

digitalWrite(ADC\_CLK,1); delayMicroseconds(2);

digitalWrite(ADC\_CLK,0);

digitalWrite(ADC\_DIO,0); delayMicroseconds(2);

digitalWrite(ADC\_CLK,1);

digitalWrite(ADC\_DIO,1); delayMicroseconds(2);

digitalWrite(ADC\_CLK,0);

digitalWrite(ADC\_DIO,1); delayMicroseconds(2);

for(i=0;i<8;i++)

{

digitalWrite(ADC\_CLK,1); delayMicroseconds(2);

digitalWrite(ADC\_CLK,0); delayMicroseconds(2);

pinMode(ADC\_DIO, INPUT);

dat1=dat1<<1 | digitalRead(ADC\_DIO);

}

for(i=0;i<8;i++)

{

dat2 = dat2 | ((uchar)(digitalRead(ADC\_DIO))<<i);

digitalWrite(ADC\_CLK,1); delayMicroseconds(2);

digitalWrite(ADC\_CLK,0); delayMicroseconds(2);

}

digitalWrite(ADC\_CS,1);

pinMode(ADC\_DIO, OUTPUT);

return(dat1==dat2) ? dat1 : 0;

}

int main(void)

{

uchar analogVal;

if(wiringPiSetup() == -1)

{

printf("setup wiringPi failed !");

return 1;

}

pinMode(ADC\_CS, OUTPUT);

pinMode(ADC\_CLK, OUTPUT);

pinMode(LedPin, OUTPUT);

while(1)

{

analogVal = get\_ADC\_Result();

printf("Current analog : %d\n", analogVal);

if(analogVal > thresholdVal)

{

digitalWrite(LedPin, HIGH);

}

else

{

digitalWrite(LedPin, LOW);

}

delay(200);

}

return 0;

}

Experimental phenomena

When the magnet is vertically close to the analog Hall sensor, a voltage difference will be generated in the analog Hall sensor. When the value of the analog Hall sensor detected by the Raspberry Pi after ADC conversion meets a certain condition, the LED lamp lights up.