Metal Touch Sensor

Overview

We will use the Raspberry Pi to control the metal touch sensor. When the analog value of the metal touch sensor suits certain conditions, the Raspberry Pi will convert the analog value to the command line interface after being converted by the ADC.

Material Needed

RaspberryPi \*1

Breadboard \*1

Metal Touch sensor \*1

ADC0832 \*1

Dupont Line

Preparatory work

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.

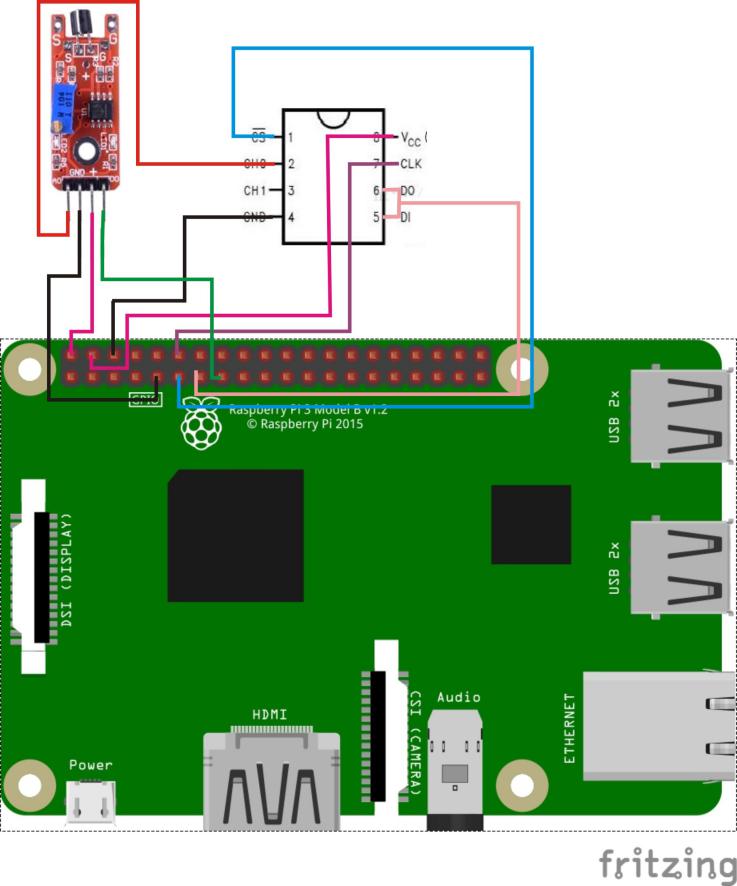
Product Description

Brief Introduction

When a metal or a human touches the touch head on a metal touch sensor, the analog output signal will change. When the analog output signal satisfies the level inversion condition, level of the digital output pin DO will flip and output high level. When the Raspberry Pi detects that the level of the digital output pin DO is inverted, it will print out the current signal value after the ADC conversion.



Wiring diagram



Sample Code

1. Python Code

#!/usr/bin/env python

import RPi.GPIO as GPIO

import ADC0832

import time

Touch\_DO\_PIN = 15

def init():

GPIO.setmode(GPIO.BOARD)

GPIO.setup(Touch\_DO\_PIN, GPIO.IN, pull\_up\_down=GPIO.PUD\_UP)

ADC0832.setup()

def loop():

print 'Please touch....\n'

while True:

global digitalVal

digitalVal = GPIO.input(Touch\_DO\_PIN)

if(digitalVal == 1):

print 'DO is %d' % digitalVal

print "Touch detected..."

print 'Current analog value is %d'% ADC0832.getResult(0)

time.sleep(0.2)

else:

pass

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 Touch\_DO\_Pin 3

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 digitalVal = 1;

uchar analogVal = 0;

if(wiringPiSetup() == -1){ //when initialize wiring failed,print messageto screen

printf("setup wiringPi failed !");

return 1;

}

pinMode(ADC\_CS, OUTPUT);

pinMode(ADC\_CLK, OUTPUT);

pinMode(Touch\_DO\_Pin, INPUT);

//pullUpDnControl(MIC\_DO\_Pin, PUD\_UP);

printf("Please touch...\n");

while(1){

if((digitalVal = digitalRead(Touch\_DO\_Pin)))

{

printf("Do is %d.\n", digitalVal);

printf("Touch detected...");

printf("Current analog value is %d.\n", get\_ADC\_Result());

delay(200);

}

else

{

;

}

}

return 0;

}

Experimental phenomena

When there is metal or human touch, if the analog signal changes satisfys the threshold condition, the corresponding value will be printed on the command line interface of the Raspberry Pi system.