```
////////////////////////////////////////////////////////////////////////////////////////////
(Zoe) Zexin Li for Liubo's final Robbie Beggar
one servo with ultrasonic distance sensor
combine the code "Sweep_two_sensors_simu" and "distance_sensor_DC_motor/example Ping"
when distance < 20cm, servo starts to goes from 100 degrees to 180 degrees
when distance> 20cm, servo doesn't move
add motion detector and buzzer
finished on April, 29, 2012
////////////////////////////////////////////////////////////////////////////////////////////

//import servo library
#include <Servo.h>
Servo first;    //define the first servo
//Servo second;

const int firstservopin = 2; // create servo object with digital pin 2
const int pingPin = 5; //create rangefinder object with digital pin 5
//const int secondservopin = 3;   //create second servo

int sensorPin = 7;   //motion detector Pin
int sensorVal, sensorValPrev;     //motion detector current value and previous value

int pos = 100; //initialize angle of servo 100 degree
int posafter = 180; //define after angle of servo 180 degree
int ledPin = 12; //LED connected to digital pin 12   ---indicates servo
int ledMotionPin = 8;   //LED connected to digital pin 8   ----indicates motion detector
int sPin = 11; //buffer connected to digital pin 11
int ledBufferPin = 4; //LED connected digital pin 11----indicate buffer

long int duration, distanceInches, distanceCm; //define time, distance "both in inch and cm"
int limitCm = 20;   //trigger point of the roration of servo

void setup()
{
pinMode(ledMotionPin, OUTPUT);    //led indicates motion detector
pinMode(sensorPin, INPUT);      //motion detector

pinMode(sPin, OUTPUT);       //buzzer
pinMode(ledBufferPin, OUTPUT);     //led indicates buzzer

sensorVal = 0;
```

```
pinMode(ledPin, OUTPUT);      //led indicates servo
first.attach(firstservopin);   //first servo attach
first.write(100); // center the servo
// second.attach(secondservopin);
// second.write(90);
delay(400);

// initialize serial communication:
Serial.begin(9600);
}

int f = 500;
int shortdelay = 10;
int fstep = 1;
long timeturnedon;

void loop()
{
//code from example of sensors_Ping,
// The PING))) is triggered by a HIGH pulse of 2 or more microseconds.
// Give a short LOW pulse beforehand to ensure a clean HIGH pulse:
pinMode(pingPin, OUTPUT); //
digitalWrite(pingPin, LOW);
delayMicroseconds(2);
digitalWrite(pingPin, HIGH);
delayMicroseconds(5);
digitalWrite(pingPin, LOW);

// The same pin is used to read the signal from the PING))): a HIGH
// pulse whose duration is the time (in microseconds) from the sending
// of the ping to the reception of its echo off of an object.
pinMode(pingPin, INPUT);
duration = pulseIn(pingPin, HIGH);

// convert the time into a distance
distanceInches = microsecondsToInches(duration);
distanceCm = microsecondsToCentimeters(duration);
checkLimit();

sensorValPrev = sensorVal;
sensorVal = digitalRead(sensorPin);
```

```
//Motion detector with led, use millis() to calculate current time, define time duration is 1s
//if sensor value equals 1, led turns on, if sensor value equals 0, led turns off.
if((sensorVal == 1 )&& (sensorValPrev == 0)){
tone(sPin, 200, 500) ; // out pin, freq (hz), dur (ms)
delay(200);
digitalWrite(ledMotionPin,HIGH);
timeturnedon = millis();
}

if(millis() > timeturnedon +1000){
noTone(sPin);
digitalWrite(ledMotionPin,LOW);
}
// Serial.print("motion detector = " );
// Serial.println(sensorVal);
// Serial print values
// Serial.print(distanceCm);
// Serial.print("cm");
// Serial.println();
}
void checkLimit()
{
if (distanceCm < limitCm){
digitalWrite(ledPin, HIGH);
for(pos = 180; pos >=100; pos-=1) // goes from 180 degrees to 0 degrees
{
first.write(pos);
first.write(100);

Serial.print(pos);
Serial.println();
// second.write(pos);
delay(40);
//Buzzer with LED
//frequency is random, led lights and off according to frequency
f = f - fstep;
//output pin is 8, 5ms delay time
f = random(900, 1500);
tone(sPin, f, shortdelay) ; // out pin, freq (hz), dur (ms)
```

```
if(f%2 == 0){
digitalWrite(ledBufferPin, HIGH);
}
if(f%2 == 1){
digitalWrite(ledBufferPin, LOW);
}
}
}
else{
first.write(180);// tell servo to go to position in variable 'pos'
delay(20); // waits 15ms for the servo to reach the position
digitalWrite(ledPin, LOW);   //led turns off when user walks away.
}
}

long microsecondsToInches(long microseconds)
{
// According to Parallax's datasheet for the PING))), there are
// 73.746 microseconds per inch (i.e. sound travels at 1130 feet per
// second).   This gives the distance travelled by the ping, outbound
// and return, so we divide by 2 to get the distance of the obstacle.
// See: http://www.parallax.com/dl/docs/prod/acc/28015-PING-v1.3.pdf
return microseconds / 74 / 2;
}

long microsecondsToCentimeters(long microseconds)
{
// The speed of sound is 340 m/s or 29 microseconds per centimeter.
// The ping travels out and back, so to find the distance of the
// object we take half of the distance travelled.
return microseconds / 29 / 2;
}
```