```
/* Rumblebot Driving with object avoidance utilizing PING Ultrasound Sensor
* and a bump switch.
* Reads values (00014-01199) from an ultrasound sensor (Parallax PING)
*
* Version 1.1.2 Motor runs and drives full time unless object detected.
*---------------------------------------------------------------------
*
*
*
*
*
*
*/
int switchPin= 2; // Right bump switch on pin 2
int swval; // Variable for reading switch status
int ultraSoundSignal = 7; // Ultrasound signal pin
int val = 0;
int ultrasoundValue = 0;
int timecount = 0; // Echo counter
int ledPin = 13; // LED connected to digital pin 13
int motorpinright = 10; // pin for left motor reverse
int motorpinleft = 11; // pin for left motor forward
int motorpinrevright = 5; // pin for right motor reverse
int motorpinrevleft = 6; // pin for right motor forward


void setup() {
pinMode(switchPin, INPUT); // Sets the digital pin as input
pinMode(ledPin, OUTPUT); // Sets the digital pin as output
pinMode(motorpinright, OUTPUT); // Motor drives----------
pinMode(motorpinleft, OUTPUT); //------------------------
pinMode(motorpinrevright, OUTPUT); //------------------------
pinMode(motorpinrevleft, OUTPUT); //------------------------
}

void loop() {



/* Start Scan
* ------------------------------------------------
*/{
timecount = 0;
val = 0;
pinMode(ultraSoundSignal, OUTPUT); // Switch signalpin to output

/* Send low-high-low pulse to activate the trigger pulse of the sensor
* ----------------------------------------------------------------
*/
```

```
digitalWrite(ultraSoundSignal, LOW); // Send low pulse
delayMicroseconds(2); // Wait for 2 microseconds
digitalWrite(ultraSoundSignal, HIGH); // Send high pulse
delayMicroseconds(5); // Wait for 5 microseconds
digitalWrite(ultraSoundSignal, LOW); // Holdoff

/* Listening for echo pulse
* ----------------------------------------------------------------
*/

pinMode(ultraSoundSignal, INPUT); // Switch signalpin to input
val = digitalRead(ultraSoundSignal); // Append signal value to val
while(val == LOW) { // Loop until pin reads a high value
val = digitalRead(ultraSoundSignal);
}

while(val == HIGH) { // Loop until pin reads a high value
val = digitalRead(ultraSoundSignal);
timecount = timecount +1; // Count echo pulse time
}

/* Lite up LED if any value is passed by the echo pulse
* ----------------------------------------------------------------
*/

if(timecount > 0){
digitalWrite(ledPin, HIGH);
delay(50); //LED on for 50 microseconds
digitalWrite(ledPin, LOW);
}

/* Delay of program
* ----------------------------------------------------------------
*/

delay(100);

}
/* Action based on data
* ----------------------------------------------------------------
*/
{
ultrasoundValue = timecount; // Append echo pulse time to ultrasoundValue
}
if (ultrasoundValue > 800)

{
/* Drive straight forward
```

```
*----------------------------------------------
*/
analogWrite(motorpinleft, 255); //100% speed
analogWrite(motorpinright, 255); //100% speed0
}
/*----------------------------------------------
*/
else
/* Turn hard right
*----------------------------------------------
*/
{
analogWrite(motorpinleft, 0); //stop left motor
analogWrite(motorpinright, 0); //stop right motor
analogWrite(motorpinrevright, 0); // stop right rev motor
analogWrite(motorpinrevleft, 0); // stop left rev motor
analogWrite(motorpinrevright, 255); //100% speed
analogWrite(motorpinleft, 255); //100% speed
delay(380); //380 milliseconds
analogWrite(motorpinrevright, 0); // off
analogWrite(motorpinleft, 0); // off

/*----------------------------------------------
*/
}

/* Backup and turn right when switch gets bumped and closes circuit to ground
*-----------------------------------------------
*/
/*
*/
digitalWrite(switchPin, HIGH); // Sets the pin to high
swval = digitalRead(switchPin); // Read input value and store it
if (swval == LOW) {
analogWrite(motorpinleft, 0); //stop left motor
analogWrite(motorpinright, 0); //stop right motor
analogWrite(motorpinrevleft, 0); // stop left rev motor
analogWrite(motorpinrevright, 0); // stop right rev motor
analogWrite(motorpinrevleft, 255); //100% speed
analogWrite(motorpinrevright, 255); //100% speed
delay(800); //800 milliseconds
analogWrite(motorpinrevleft, 0); // off
analogWrite(motorpinrevright, 0); // off
analogWrite(motorpinrevright, 255); //100% reverse speed
analogWrite(motorpinleft, 255); //100% forward speed
delay(700); //700 milliseconds
analogWrite(motorpinrevright, 0); // off
analogWrite(motorpinleft, 0); // off
delay(50); //50 milliseconds
```

```
}

/*-----------------------------------------
*/


}
```