

```

// this program demonstrates the communication from the control laptop
// to the arduino and the commands to steer two DC motors
//
// you need an USB cable to program the arduino
// pin number of on-board LED
int ledPin = 13;

// Pulse Width Modulation (PWM) pins
int PWM1 = 3; //LV
int PWM2 = 5; //LA
int PWM3 = 6; //RV
int PWM4 = 11; //RA
// light pin
//int LIT = 2;
// Sensor in Arduino

const int trigPin1 = 7;
const int echoPin1 = 8;
long duration1;
int distanceCm1 = 2;
int distance1;
const int trigPin2 = 12;
const int echoPin2 = 13;
long duration2;
int distanceCm2 = 7;
int distance2;
int n;
int t;
int a;
char Command;

void setup() {

  // Setup Sonar Sensors -----
  pinMode(trigPin1, OUTPUT);
  pinMode(echoPin1, INPUT);
  pinMode(trigPin2, OUTPUT);
  pinMode(echoPin2, INPUT);
  // setup DC motors -----
  // all outputs to zero
  analogWrite(PWM1,0);
  analogWrite(PWM2,0);
  analogWrite(PWM3,0);
  analogWrite(PWM4,0);
  digitalWrite(ledPin, HIGH);

  Serial1.begin(115200); //for Ethernet or Wifi

  // clear the input buffer
  while (Serial1.available())
    Serial1.read();

  };
void loop() {
  measure();
  if(Serial1.available() > 0)
  {
    Command = Serial1.read();

    if(Command == '5')
    {Serial1.print("\n");
      Serial1.print("You stopping?: ");
      pause();
    }

    else if(Command == '8')

```

```

{
while(distanceCm2 <10 && distanceCm2 > 0.1 && Command == '8') //check if the distance to the floor is right
{input();
measure();
Serial1.print("\n");
Serial1.print(" You are not approaching the end of the platform, going forward:");
if(distanceCm1 > 70 || distanceCm1 == 0) // If the object is far away or not existing go on
{
while((distanceCm1 > 70 || distanceCm1 == 0) && Command =='8')
{
input();
measure();
forward();
}
}
if(distanceCm1 > 40 && distanceCm1 <70)
{
Serial1.print("\n");
Serial1.print(" obstacle in less than 40cm, slowing down:");
while((distanceCm1 > 40 && distanceCm1 < 70) && Command == '8') // If the object is not to close continue slowly
{
input();
measure();
analogWrite(PWM1,55);
analogWrite(PWM2,0);
analogWrite(PWM3,55);
analogWrite(PWM4,0);
}
}
n=0;
if(distanceCm1 < 40)// stop if the object is too close
{
Serial1.print("\n");
Serial1.print(" obstacle is getting too close, evading obstacle: ");
while(distanceCm1<40 && Command == '8') //make a left turn until the object is gone again and count n
{
input();
measure();

leftturn();
n=n+1;
delay(400);
}
}

if(distanceCm1 > 40 || distanceCm1 == 0)
{
Serial1.print("\n");
Serial1.print(" obstacle is gone, driving forward ");
a=0;
while(a<3 && (distanceCm1 > 40 || distanceCm1 == 0) && Command == '8') //when the object is gone drive forward
{
input();
measure();
forward();
delay(400);
a=a+1;
Serial1.print("\n");
Serial1.print(3-a);
}
t=0;
Serial1.print(n);
Serial1.print("\n");
Serial1.print(t);
if(distanceCm1 >40 || distanceCm1==0)
{

```

```

t=0;
Serial1.print("\n");
Serial1.print(" obstacle is gone, turning back ");
while((distanceCm1 > 40 || distanceCm1 == 0) && t<n && Command == '8') //make a right turn back and count t, when n is equal to t stop
turning, same angle should be traveled
{
  input();
  measure();
  t=t+1;
  rightturn();
  delay(400);
}
}
if(distanceCm1 > 40 || distanceCm1 == 0)
{
  Serial1.print("\n");
  Serial1.print(" obstacle is gone, driving forward ");
  a=0;
  while(a<6 && (distanceCm1 > 40 || distanceCm1 == 0) && Command == '8') //when the object is gone drive forward
  {
    input();
    measure();
    forward();
    delay(400);
    a=a+1;
    Serial1.print("\n");
    Serial1.print(6-a);
  }
}
if(distanceCm1 >40 || distanceCm1==0)
{
  t=0;
  Serial1.print("\n");
  Serial1.print(" obstacle is gone, turning back right ");
  while((distanceCm1 > 40 || distanceCm1 == 0) && t<n && Command == '8') //make a right turn back and count t, when n is equal to t stop
turning, same angle should be traveled
  {
    input();
    measure();
    t=t+1;
    rightturn();
    delay(400);
  }
}
if(distanceCm1 > 40 || distanceCm1 == 0)
{
  Serial1.print("\n");
  Serial1.print(" obstacle is gone, driving forward ");
  a=0;
  while(a<5 && (distanceCm1 > 40 || distanceCm1 == 0) && Command == '8') //when the object is gone drive forward
  {
    input();
    measure();
    forward();
    delay(400);
    a=a+1;
    Serial1.print("\n");
    Serial1.print(5-a);
  }
}
if(distanceCm1 >40 || distanceCm1==0)
{
  t=0;
  Serial1.print("\n");
  Serial1.print(" obstacle is gone, turning back left ");

```

```

    while((distanceCm1 > 40 || distanceCm1 == 0) && t < n && Command == '8') //make a right turn back and count t, when n is equal to t stop
turning, same angle should be traveled
    {
        input();
        measure();
        t=t+1;
        leftturn();
        delay(20);
    }

}

if(distanceCm2 > 200 || distanceCm2 < 0.1) // if distance to the floor is too large stop driving, end has been reached
{
    Serial1.print(" You are approaching the end of the platform, i'm going to stop now.");
    pause();
}
}

else if(Command == '7')
{

    Serial1.print("GEAR UP");
    analogWrite(PWM1,150);
    analogWrite(PWM2,0);
    analogWrite(PWM3,150);
    analogWrite(PWM4,0);
}

else if(Command == '9')
{

    Serial1.print("GOING IN THE FAST LANE");
    analogWrite(PWM1,255);
    analogWrite(PWM2,0);
    analogWrite(PWM3,255);
    analogWrite(PWM4,0);
}

else if(Command == '4') //controls for user
{
    Serial1.print("You're going left");
    leftturn();
}

else if(Command == '6')
{
    Serial1.print("You're going right");
    rightturn();
}
}

}

void input()
{
    if(Serial1.available() > 0)
    {
        Command = Serial1.read();
    }
}

void rightturn() {
    analogWrite(PWM1,230);
    analogWrite(PWM2,0);
    analogWrite(PWM3,0);
    analogWrite(PWM4,255);
}

void leftturn() {
    analogWrite(PWM1,0);
    analogWrite(PWM2,255);
    analogWrite(PWM3,255);
    analogWrite(PWM4,0);
}

```

```

}
void forward() {
  analogWrite(PWM1,190);
  analogWrite(PWM2,0);
  analogWrite(PWM3,255);
  analogWrite(PWM4,0);
}
void pause() {
  analogWrite(PWM1,0);
  analogWrite(PWM2,0);
  analogWrite(PWM3,0);
  analogWrite(PWM4,0);
}
void measure() {
  digitalWrite(trigPin1, LOW);
  delayMicroseconds(1);
  digitalWrite(trigPin1, HIGH);
  delayMicroseconds(15);
  digitalWrite(trigPin1, LOW);
  duration1 = pulseIn(echoPin1, HIGH);
  distanceCm1 = duration1*0.034/2;

  delay(10);

  //digitalWrite(trigPin2, LOW);
  //delayMicroseconds(1);
  //digitalWrite(trigPin2, HIGH);
  //delayMicroseconds(15);
  //digitalWrite(trigPin2, LOW);
  //duration2 = pulseIn(echoPin2, HIGH);
  //distanceCm2 = duration2*0.034/2;
  //delay(100);

  //Serial1.print(" Distance1: ");
  //Serial1.println(distanceCm1);
  //Serial1.print(" Distance2: ");
  //Serial1.println(distanceCm2);
}

```