

```

1. #include <Wire.h>
2. #include <LiquidCrystal_I2C.h>
3.
4. LiquidCrystal_I2C lcd(0x27,16,2);  //
5.
6. //Pines de los Motores
7. int IN1 = 7;          // IN1 de L298N a pin digital 2
8. int IN2 = 8;          // IN2 de L298N a pin digital 3
9. int ENA = 9;          // ENA de L298N a pin digital 5
10. int IN3 =13;
11. int IN4 =12;
12. int ENB=11;
13.
14. //Pines para sensores
15. //Centro
16. const int Trigger_centro = 4;
17. const int Echo_centro = 5;
18.
19. //Izquierda
20. const int Trigger_izq = 2;
21. const int Echo_izq = 3;
22.
23. //Derecha
24. const int Trigger_der = 6;
25. const int Echo_der = 10;
26.
27. //Pines infrarrojos
28. const int sensorPin_izq=A2;
29. const int sensorPin_der = A3;
30.
31.
32. void setup(){
33.   Serial.begin(9600);
34.   //Inicializacion Motores
35.   pinMode(IN1, OUTPUT);  // pin 2 como salida
36.   pinMode(IN2, OUTPUT);  // pin 3 como salida
37.   pinMode(ENA,OUTPUT);
38.   pinMode(IN3, OUTPUT);  // pin 2 como salida
39.   pinMode(IN4, OUTPUT);  // pin 3 como salida
40.   pinMode(ENB,OUTPUT);
41.   // pin 5 como salida
42.
43.   //Inicializacion de la pantalla LCD
44.   lcd.init();
45.
46.   //Encender la luz de fondo.
47.   lcd.backlight();
48.

```

```

49. // Escribimos el Mensaje en el LCD.
50. lcd.print("Hola Mundo");
51. lcd.clear();
52.
53. //Inicializacion Sensores
54. pinMode(Triquer_centro, OUTPUT); //pin como salida
55. pinMode(Echo_centro, INPUT); //pin como entrada
56. digitalWrite(Triquer_centro, LOW); //Inicializamos el pin
    con 0
57.
58. pinMode(Triquer_izq, OUTPUT); //pin como salida
59. pinMode(Echo_izq, INPUT); //pin como entrada
60. digitalWrite(Triquer_izq, LOW); //Inicializamos el pin con
    0
61.
62. pinMode(Triquer_der, OUTPUT); //pin como salida
63. pinMode(Echo_der, INPUT); //pin como entrada
64. digitalWrite(Triquer_der, LOW); //Inicializamos el pin con
    0
65.
66. //Inicializamos infrarrojos
67. pinMode(sensorPin_izq , INPUT); //definir pin como
    entrada
68. pinMode(sensorPin_der , INPUT); //definir pin como
    entrada
69.
70. delay(1500);
71.
72. }
73.
74. void loop(){
75.     bool accion = false;
76.     int valor_ultrasonidos = 0;
77.     int value_izq = 0;
78.     int value_der = 0;
79.     value_izq = digitalRead(sensorPin_izq ); //lectura
        digital de pin
80.     value_der = digitalRead(sensorPin_der ); // lectura
        digital del p
81.
82.     //Motor avanza
83.     motorAvanza();
84.
85.     lcd.clear();
86.     if (value_izq == 0 && !accion) {
87.         Serial.println("Detectado obstaculo izq");
88.         lcd.print("Linea izq");
89.         motorGiraDer();

```

```

90.         accion = true;
91.         delay(1000);
92.     }
93.     lcd.clear();
94.     if (value_der == 0 && !accion) {
95.         Serial.println("Detectado obstaculo der");
96.         lcd.print("Linea der");
97.         motorGiraIzq();
98.         accion = true;
99.         delay(1000);
100.
101.     }
102.     lcd.clear();
103.     lcd.setCursor(0,0);
104.
105.     valor_ultrasonidos = sensores();
106.     if(valor_ultrasonidos == 0){
107.
108.         lcd.print("ataca delan");
109.         motorAvanza();
110.         delay(500);
111.     }
112.     if(valor_ultrasonidos == 2){
113.         lcd.print("ataca der");
114.         motorGiraDer();
115.         delay(500);
116.     }
117.     if(valor_ultrasonidos == 1){
118.         lcd.print("ataca izq");
119.         motorGiraIzq();
120.         delay(500);
121.     }
122.
123.
124. }
125.
126. void motorAvanza(){
127.
128.
129.     //Motor Izquierdo
130.     digitalWrite(IN1, LOW);           // IN1 en 1
131.     digitalWrite(IN2, HIGH);
132.
133.
134.     analogWrite(ENA, 255);
135.         // el valor de velocidad y aplica a ENA
136.
137.     digitalWrite(IN3, LOW);           // IN1 en 1

```

```

138.  digitalWrite(IN4, HIGH); // IN2 en 0
139.  analogWrite(ENB, 255);
140. }
141.
142. void motorGiraIzq(){
143.
144.
145.  digitalWrite(IN1, HIGH);          // IN1 en 1
146.  digitalWrite(IN2, LOW);
147.
148.
149.  analogWrite(ENA, 190);
150.      // el valor de velocidad y aplica a ENA
151.
152.  digitalWrite(IN3, LOW);          // IN1 en 1
153.  digitalWrite(IN4, LOW); // IN2 en 0
154.  analogWrite(ENB, 190);
155. }
156.
157. void motorGiraDer(){
158.
159.  digitalWrite(IN1, LOW);          // IN1 en 1
160.  digitalWrite(IN2, LOW);
161.
162.
163.  analogWrite(ENA, 190);
164.      // el valor de velocidad y aplica a ENA
165.
166.  digitalWrite(IN3, HIGH);          // IN1 en 1
167.  digitalWrite(IN4, LOW); // IN2 en 0
168.  analogWrite(ENB, 190);
169. }
170.
171. int sensores(){
172.  long t_centro, t_izq, t_der, t_min1, t_min2; //timepo que
    demora en llegar el eco
173.  long d; //distancia en centimetros
174.
175.  int centro = 0, izquierda = 1, derecha =2, devolver = -1;
176.
177.  //Distancia centro
178.  digitalWrite(Triquer_centro, HIGH);
179.  delayMicroseconds(10);          //Enviamos un pulso de
    10us
180.  digitalWrite(Triquer_centro, LOW);
181.
182.  t_centro = pulseIn(Echo_centro, HIGH); //obtenemos el
    ancho del pulso

```

```

183.
184. //Distancia izq
185. digitalWrite(Trigger_izq, HIGH);
186. delayMicroseconds(10);          //Enviamos un pulso de
    10us
187. digitalWrite(Trigger_izq, LOW);
188.
189. t_izq = pulseIn(Echo_izq, HIGH); //obtenemos el ancho del
    pulso
190.
191. //Distancia der
192. digitalWrite(Trigger_der, HIGH);
193. delayMicroseconds(10);          //Enviamos un pulso de
    10us
194. digitalWrite(Trigger_der, LOW);
195.
196. t_der = pulseIn(Echo_der, HIGH); //obtenemos el ancho del
    pulso
197.
198. t_min1 = min(t_centro, t_izq);
199. t_min2 = min(t_min1, t_der);
200.
201. if(t_min2 == t_centro){
202.     devolver = centro;
203.     Serial.println("Centro");
204. }
205. if(t_min2 == t_der){
206.     devolver = derecha;
207.     Serial.println("Derecha");
208. }
209. if(t_min2 == t_izq){
210.     devolver = izquierda;
211.     Serial.println("Izquierda");
212. }
213.
214.
215. d = t_min2/59;                    //escalamos el tiempo a una
    distancia en cm
216. if(d < 40){
217.     return devolver;
218. }
219. else{
220.     devolver = -1;
221.     return devolver;
222. }
223. /* Serial.print("Distancia: ");
224.     Serial.print(d);              //Enviamos serialmente el valor de
    la distancia

```

```
225.  Serial.print("cm");
226.  Serial.println();
227.  delay(100);          */    //Hacemos una pausa de 100ms*/
228. }
```