



INSTITUTO TECNOLÓGICO SUPERIOR DE ALAMO TEMAPACHE

REPORTE DE PRÁCTICA

No. De la práctica:	05
Fecha:	16/12/2023
Lugar:	Laboratorio
Páginas:	Página 1 de 9

DATOS GENERALES

NOMBRE DE LA PRÁCTICA:	PROGRAMACIÓN MICROCONTROLADORES CARRITO CON SENSORES
ASIGNATURA:	SISTEMAS PROGRAMABLES
UNIDAD TEMÁTICA:	U5 PUERTOS Y BUSES COMUNICACIÓN
SNOMBRE DEL DOCENTE:	SILVERIO RAMÍREZ MARTÍNEZ
INTEGRANTES DEL EQUIPO:	GONZALO MARTINEZ SILVERIO MARIA CRUZ GOMEZ SANDRA LIZBETH MARTINEZ MTZ RUTH YZELDA YAMILETH VAZQUEZ RAMOS
GRUPO:	701ª
CARRERA:	INGENIERIA EN SISTEMAS COMPUTACIONALES



INSTITUTO TECNOLÓGICO SUPERIOR DE ALAMO TEMAPACHE

INTRODUCCIÓN

La programación de microcontroladores en la construcción de carritos autónomos con sensores representa un emocionante desafío en el campo de la ingeniería y la robótica. Estos diminutos dispositivos electrónicos ofrecen un vasto espectro de posibilidades, permitiendo la creación de sistemas inteligentes capaces de interpretar su entorno y responder de manera autónoma. En este contexto, el diseño de carritos impulsados por microcontroladores y equipados con sensores se ha vuelto un área de gran interés, ya que ofrece aplicaciones prácticas en la automatización industrial, la logística, la domótica y otros campos.

OBJETIVO

Investigar la relación entre distintos puertos de comunicación y sus aplicaciones

COMPETENCIA A DESARROLLAR

Identifica y analiza los elementos esenciales de los puertos y buses de comunicación

Implementa las aplicaciones que impliquen el manejo de puertos y buses de comunicación

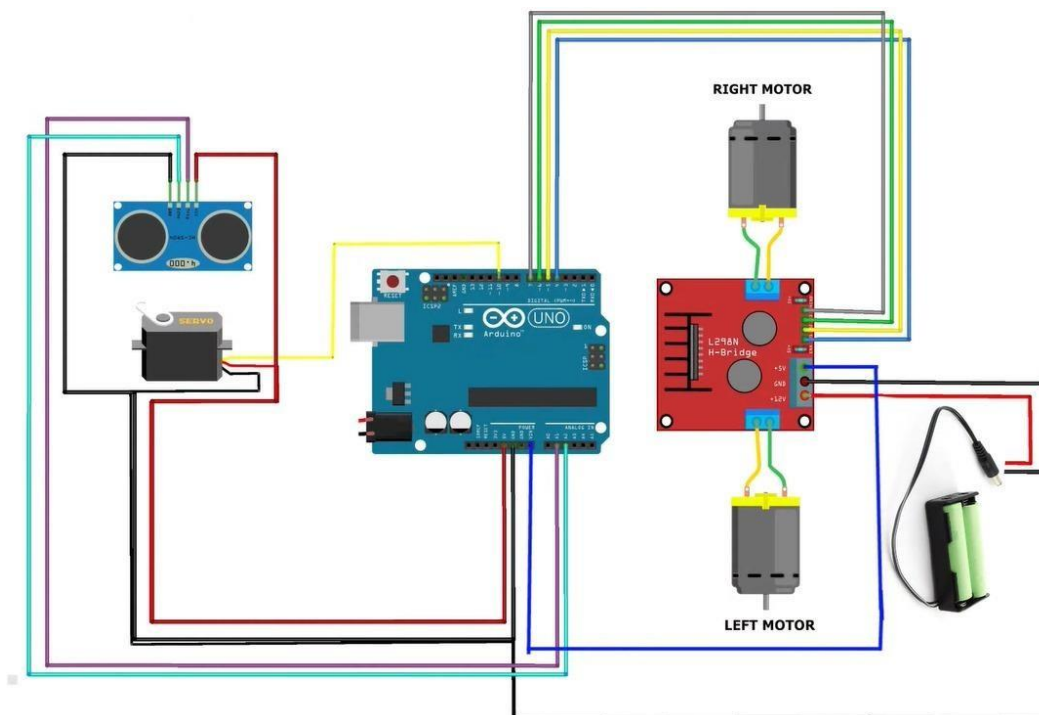
MATERIAL Y EQUIPO

- Laptop
- Arduino IDE.
- Tinkercad
- Circuitos eléctricos
- Sensores
- Microcontrolador

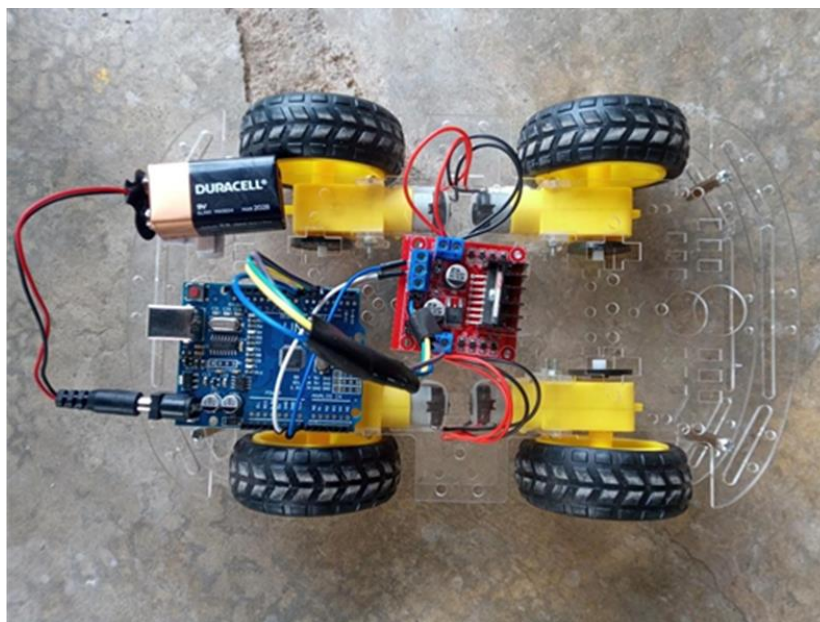


INSTITUTO TECNOLÓGICO SUPERIOR DE ALAMO TEMAPACHE

1. Diseñamos nuestro carro:



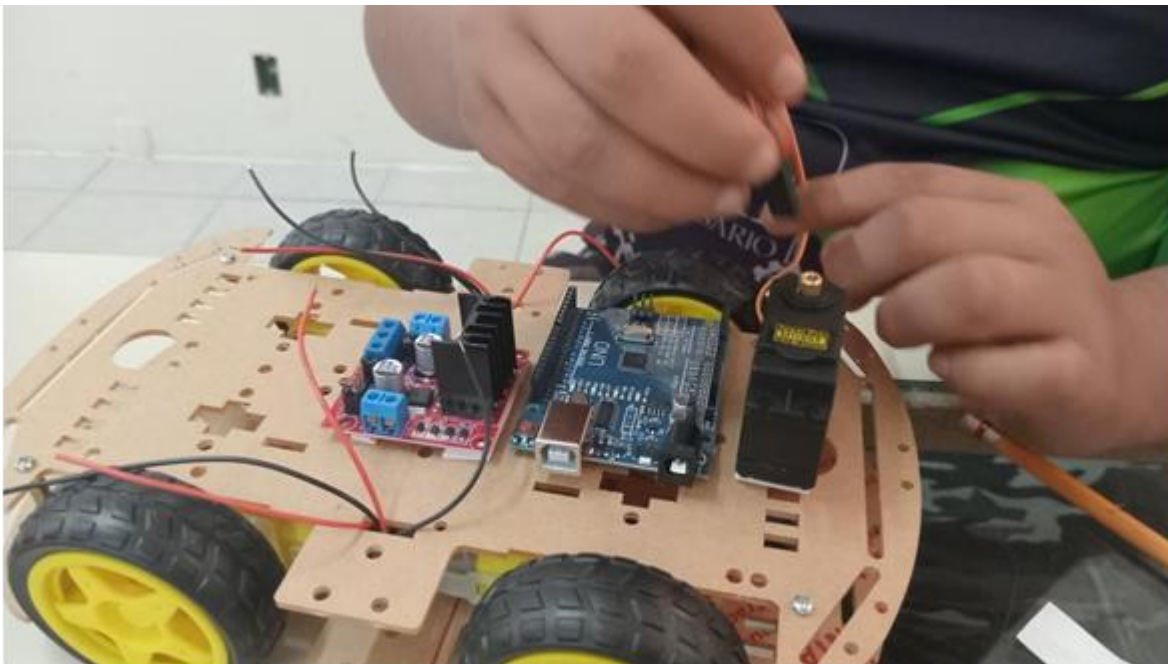
2. Reutilizamos el carro de la practica anterior para proseguir con la actual debido a que ya se contaba con el avance.





INSTITUTO TECNOLÓGICO SUPERIOR DE ALAMO TEMAPACHE

3. Agregamos un servomotor y el sensor ultrasónico.





INSTITUTO TECNOLÓGICO SUPERIOR DE ALAMO TEMAPACHE

4. Así quedará al final



5. Conectamos el Arduino a la computadora, seleccionamos el puerto y la placa.





6. Comenzamos a codificar lo que queremos que haga el carro.

```
carro2 | Arduino IDE 2.2.1
Archivo  Editar  Sketch  Herramientas  Ayuda

[Check] [Run] [Upload] [Serial] [Arduino Uno]

carro2.ino
1  #include <Servo.h>
2  #include <NewPing.h>
3
4  // Definición de pines para el control de motores
5  const int MotorIzquierdoAdelante = 7;
6  const int MotorIzquierdoAtras = 6;
7  const int MotorDerechoAdelante = 5;
8  const int MotorDerechoAtras = 4;
9
10 // Definición de pines para el sensor ultrasónico
11 #define pin_trig A1
12 #define pin_echo A2
13 #define distancia_maxima 200
14
15 boolean avanza = false; // Variable para controlar la dirección del movimiento
16 int distancia = 100; // Distancia inicial
17
18 NewPing sonar(pin_trig, pin_echo, distancia_maxima);
19 Servo motor_servo;
20
21 void setup() {
22   // Configuración de pines de salida para el control de motores
23   pinMode(MotorDerechoAdelante, OUTPUT);
24   pinMode(MotorIzquierdoAdelante, OUTPUT);
25   pinMode(MotorIzquierdoAtras, OUTPUT);
26   pinMode(MotorDerechoAtras, OUTPUT);
27
28   // Inicialización del servo motor
29   motor_servo.attach(10);
30
31   // Posicionamiento inicial del servo y lectura de distancia
32   motor_servo.write(115);
33   for (int i = 0; i < 4; ++i) {
34     delay(2000);
35     distancia = leerDistancia();
36     delay(100);
37   }
38 }
39
40 void loop() {
41   delay(50);
42   distancia = leerDistancia();
43
44   if (distancia <= 45) {
45     detenerMovimiento();
46     delay(200);
47     retroceder();
48     delay(400);
49     detenerMovimiento();
50     delay(200);
51     int distanciaDerecha = mirarDerecha();
52     delay(200);
53     int distanciaIzquierda = mirarIzquierda();
54     delay(200);
55   }
```



```
56 // Gira hacia la dirección con mayor espacio libre
57 if (distancia >= distanciaIzquierda) {
58     girarDerecha();
59     detenerMovimiento();
60 } else {
61     girarIzquierda();
62     detenerMovimiento();
63 }
64 } else {
65     avanzar();
66 }
67 }
68
69 // Gira el servo a la derecha y devuelve la distancia actual
70 int mirarDerecha() {
71     motor_servo.write(50);
72     delay(500);
73     int distancia = leerDistancia();
74     delay(100);
75     motor_servo.write(115);
76     return distancia;
77 }
78
79 // Gira el servo a la izquierda y devuelve la distancia actual
80 int mirarIzquierda() {
81     motor_servo.write(170);
82     delay(500);
83     int distancia = leerDistancia();
84     delay(100);
85     motor_servo.write(115);
86     return distancia;
87 }
88
89 // Lee la distancia actual desde el sensor ultrasónico
90 int leerDistancia() {
91     delay(70);
92     int cm = sonar.ping_cm();
93     if (cm == 0) {
94         cm = 250; // Valor de distancia máximo en caso de error
95     }
96     return cm;
97 }
98
99 // Detiene todos los motores
100 void detenerMovimiento() {
101     digitalWrite(MotorDerechoAdelante, LOW);
102     digitalWrite(MotorIzquierdoAdelante, LOW);
103     digitalWrite(MotorDerechoAtras, LOW);
104     digitalWrite(MotorIzquierdoAtras, LOW);
105 }
106
```



```
107 // Mueve hacia adelante
108 void avanzar() {
109     if (lavanza) {
110         avanza = true;
111         digitalWrite(MotorIzquierdoAdelante, HIGH);
112         digitalWrite(MotorDerechoAdelante, HIGH);
113         digitalWrite(MotorIzquierdoAtras, LOW);
114         digitalWrite(MotorDerechoAtras, LOW);
115     }
116 }
117
118 // Mueve hacia atrás
119 void retroceder() {
120     avanza = false;
121     digitalWrite(MotorIzquierdoAtras, HIGH);
122     digitalWrite(MotorDerechoAtras, HIGH);
123     digitalWrite(MotorIzquierdoAdelante, LOW);
124     digitalWrite(MotorDerechoAdelante, LOW);
125 }
126
127 // Gira a la derecha
128 void girarDerecha() {
129     digitalWrite(MotorIzquierdoAdelante, HIGH);
130     digitalWrite(MotorDerechoAtras, HIGH);
131     digitalWrite(MotorIzquierdoAtras, LOW);
132     digitalWrite(MotorDerechoAdelante, LOW);
133     delay(400);
134     digitalWrite(MotorIzquierdoAdelante, HIGH);
135     digitalWrite(MotorDerechoAdelante, HIGH);
136     digitalWrite(MotorIzquierdoAtras, LOW);
137     digitalWrite(MotorDerechoAtras, LOW);
138 }
139
140 // Gira a la izquierda
141 void girarIzquierda() {
142     digitalWrite(MotorIzquierdoAtras, HIGH);
143     digitalWrite(MotorDerechoAdelante, HIGH);
144     digitalWrite(MotorIzquierdoAdelante, LOW);
145     digitalWrite(MotorDerechoAtras, LOW);
146     delay(400);
147     digitalWrite(MotorIzquierdoAdelante, HIGH);
148     digitalWrite(MotorDerechoAdelante, HIGH);
149     digitalWrite(MotorIzquierdoAtras, LOW);
150     digitalWrite(MotorDerechoAtras, LOW);
151 }
152
153
```




CONCLUSIONES

En conclusión, la programación de microcontroladores para carritos con sensores no solo representa un avance tecnológico en la robótica, sino que también abre nuevas fronteras en términos de eficiencia y automatización. La capacidad de estos diminutos dispositivos para interpretar señales del entorno a través de sensores y traducirlas en acciones específicas demuestra el poder de la programación y la electrónica en conjunto. A medida que avanzamos hacia un futuro cada vez más automatizado, la habilidad de programar microcontroladores para crear sistemas autónomos se vuelve esencial para mejorar la eficiencia en diversos campos de aplicación.

Bibliografía

Microcontroladores-Pic - Mikro. (S. F.). Mikro. Programacion-En-C-Con-
<https://www.Mikroe.Com/Ebooks/Microcontroladores-Pic-Ejemplos/Microcontroladores-Pic> Silverio, Y. A. C. (2017,

16 Mayo). Microcontroladores Pic. Monografias.Com.
<https://www.Monografias.Com/Docs113/Microcontroladores-Pic/Microcontroladores-P>