

SISTEMA DE AUTOMATIZACIÓN CON ARDUINO PARA  
EL CONTROL DE SALIDA Y LLEGADA DE LOS VEHÍCULOS  
DE UN PARQUEADERO

*Automation system with arduino for the control of  
departure and arrival of vehicles  
from a parking lot*

Cristian cortes

John solarte

Ingeniería Informática, Facultad de Ingeniería.

Institución Universitaria Colegio Mayor del Cauca

Popayán, Colombia

**Resumen:** El creciente aumento de la población junto con la utilización de vehículos se ve reflejada en un mayor tránsito vehicular y mayor requerimiento de plazas de estacionamiento, por lo que se necesita soluciones enfocadas a la movilidad y gestión eficiente de parqueaderos, para esto se presentan diferentes soluciones tecnológicas de bajo costo y de código abierto como lo es la plataforma Arduino, que es una potente Plataforma que puede dar solución a diferentes tipos de problemas, como la automatización de un parqueadero utilizando dicha plataforma y diferentes sensores y actuadores que nos permitirán controlar un paso de vehículos de un parqueadero.

**Palabras claves:** soluciones tecnológicas, Arduino, código abierto

### **Abstract**

The growing increase in the population together with the use of vehicles is reflected in a greater vehicular traffic and greater requirement for parking spaces, which is why solutions focused on mobility and efficient parking management are needed, for which different solutions are presented. low-cost and open source technology such as the Arduino platform, which is a powerful platform that can provide solutions to different types of problems, such as the automation of a parking lot using said platform and different sensors and actuators that will allow us to control a step of vehicles from a parking lot, in order to save time and money for the owners of the establishment

### **Introducción**

La movilidad en los sectores más concurridos de la ciudad de Popayán, se ve afectada hoy en día, por los conductores que estacionan sus vehículos en los costados de las calles, generando un colapso dado que no se están utilizando en su totalidad los carriles destinados para el tráfico vehicular; son múltiples las

causas de este problema, pero una de las más significativas es el elevado costo de las tarifas que tienen los parqueaderos privados, por lo cual los conductores optan por estacionar en las calles sus vehículos.

En este documento se plantea el diseño de una arquitectura inteligente, para la gestión de un parqueadero, que utilizara tecnologías de bajo costo, una posible a optima solución para este problema es la plataforma Arduino.

Arduino es una compañía de desarrollo de software y hardware libres, así como una comunidad internacional que diseña y manufactura placas de desarrollo de hardware para construir dispositivos digitales y dispositivos interactivos que puedan detectar y controlar objetos del mundo real. Arduino se enfoca en acercar y facilitar el uso de la electrónica y programación de sistemas embebidos en proyectos multidisciplinarios.

Por esta razón se presentará una clara solución al problema visto anteriormente, además de esto la plataforma Arduino permite una gran compatibilidad con diferentes dispositivos como pueden ser los sensores y actuadores, que permiten dar una visión en tiempo real de lo que esta pasando en un sistema.

### **Objetivo general**

automatización de un parqueadero con Arduino para el control de salida y llegada de los vehículos

### **objetivos específicos**

1. Diseñar la arquitectura SBC adecuada para el control de espacios disponibles en el parqueadero
2. Desarrollar una Maqueta donde se muestre el funcionamiento a escala del sistema
3. escoger los dispositivos hardware y software adecuados de la infraestructura del sistema completo

### **Trabajos relacionados**

El presente proyecto busca el aprovechamiento de las tecnologías inteligentes de la plataforma Arduino que utilicen los sensores y actuadores dirigidos para la gestión de un parqueadero.

Los siguientes proyectos presentaran algunas soluciones para la gestión de un parqueadero inteligente:

*Osorio, F. G., García, R. E. & Rincón, A. F.*

*(2013). Parqueadero inteligente. Recuperado de: <http://hdl.handle.net/20.500.12749/15402>.*

En este documento se detallan el proceso de diseño y construcción de un modelo de parqueadero inteligente, que proveerá un servicio de estacionamiento automatizado. El aspecto más relevante de nuestro proyecto es que puede ser implementado en espacios limitados y tener la mayor cantidad de estacionamientos posibles. El resultado también incluirá un sistema de 'autoparking' en el que no se necesitará intervención mayor del operario y ninguna del usuario sin malgastar tiempo en buscar un espacio vacío

*Cubillos-García, N. & Rodríguez-Mora, J. S. (2018). Arquitectura IoT para parqueaderos inteligentes en la ciudad de Bogotá. Trabajo de Grado. Universidad Católica de Colombia. Facultad de Ingeniería. Programa de Ingeniería de Sistemas. Bogotá, Colombia*

En el proyecto de grado se establece un modelo de arquitectura IoT para un sistema de parqueaderos inteligentes en la ciudad de Bogotá, dicha arquitectura fue generada bajo el marco de referencia establecido por TOGAF, estableciendo así sus diferentes componentes como arquitectura de aplicación, infraestructura, datos y negocio. Para establecer esta arquitectura se realizó una comparación de las diferentes arquitecturas y modelos que han sido implementadas y se encuentran funcionando actualmente en varios países alrededor del mundo. Dichos modelos fueron consultados en las páginas web de los ayuntamientos y alcaldías de cada ciudad y también de las páginas de los proveedores y contratistas de estos servicios para las ciudades. El objetivo de este proyecto es generar un modelo de arquitectura aplicable y efectivo en las bahías de parqueo de la ciudad de Bogotá permitiendo así que la ciudad dé el primer paso para convertirse en una ciudad inteligente implementando este tipo de tecnologías. Como muestra de la arquitectura se establece un prototipo donde se observan los componentes descritos a lo largo del documento, este prototipo sirve como base para validar

la viabilidad de la arquitectura de ser implementada en la ciudad de Bogotá.

## **Estado del arte**

Por un momento pensemos la cantidad de espacio que ocupan los parqueaderos de los automóviles en las distintas ciudades de mundo, campos de parqueo tan grandes como una cancha de futbol, si tal vez los parqueaderos se reubicaran y estos grandes espacios se les dejaran a zonas verdes los efectos de los cambios climáticos cambiarían. Una ventaja fundamental de este tipo de aparcamientos es la reducción de vehículos en la vía pública con el consiguiente ahorro de emisiones de CO<sub>2</sub>. Además, este sistema evita hurtos, ya que ninguna persona puede entrar en el espacio donde se ubican los vehículos.

Existen muchos tipos de parqueaderos automatizados como los son parqueadero de rompecabezas o puzzle parking 2 [Parking Management International, 2009] Con los años, los sistemas de aparcamiento y las tecnologías que la acompañan han incrementado y diversificado. Los sistemas de plazas de estacionamiento han existido desde hace tiempo ya que los autos fueron inventados.

En cualquier área donde hay una cantidad significativa de tráfico, hay sistemas de aparcamiento. Los sistemas de plazas de aparcamiento se desarrollaron en el siglo 20 en respuesta a la necesidad de espacio de almacenamiento para los vehículos. En la década de 1920, precursores de los sistemas de aparcamiento automático apareció en ciudades estadounidenses como Los Ángeles, Chicago, Nueva York y Cincinnati. Algunas de estas estructuras de varios pisos se mantienen en pie, y se han adaptado para nuevos usos. Uno de los aparcamientos automáticos Kent en Nueva York es un hito del arte, que fue convertido en lujosos condominios en 1983.

Un sistema que ahora se encuentran por todo Japón fue creado por la Corporación Westinghouse en 1923 y, posteriormente, construido en 1932 en Monroe Street de Chicago.

La Compañía Nash Motor creó la versión primera copa cerrada de este sistema para el Chicago siglo de progreso de Exposiciones en 1933, y fue el precursor de

una versión más reciente, las Torres de coches inteligentes en Europa. 3[Sanders, 2010] El parqueadero de rompecabezas es un sistema de aparcamiento en una combinación de movimiento horizontal y vertical de las plataformas y tarimas que se llevan a cabo para dar cabida a un mayor número de coches en el espacio disponible.

### Metodología del diseño

Para el diseño del sistema, se establece una metodología de detalle el cual se basa básicamente en el planteamiento del problema, el planteamiento de posibles soluciones, el uso de actuadores, sensores, controladores, desarrollo de simulaciones del sistema y elaboración del prototipo. Planteamiento del problema: en esta etapa del diseño observamos y planteamos todos los posibles problemas directos o indirectos que pueden ser ocasionados por el uso de parqueaderos comunes. Planteamiento de posibles soluciones: acá empezamos a dar ideas de posibles soluciones y luego escogimos las más posibles y más llamativas.

*Actuadores:* en la etapa de los actuadores escogimos varios tipos de actuadores considerando su requerimiento su precio y disponibilidad

*Sensores:* en la etapa de los sensores escogimos varios tipos de sensores considerando su requerimiento su precio y disponibilidad.

### Montaje del sistema

Para el montaje del sistema se utilizan los siguientes materiales para dar una visión mejorada del sistema

*sensor laser ky-008:* El láser emite un rayo láser hacia el objeto como se muestra arriba. La luz reflejada por el objeto es concentrada por el lente del receptor

*sensor de luz ky-018:* Un sensor de luminosidad es un dispositivo que permite conocer el grado de iluminación de un entorno concreto

*push button:* Utilizado como switch o interruptor al momento de ser presionado. Funciona como contacto normalmente abierto

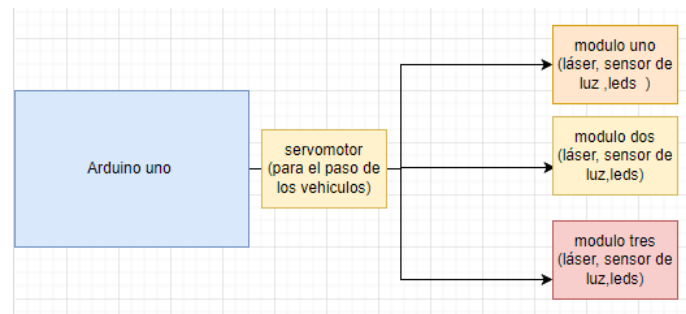
*servo motor sg-90:* Un servomotor es un actuador rotativo o motor que permite un control preciso en términos de posición angular, aceleración y velocidad, capacidades que un motor normal no tiene.

*Arduino uno:* El **Arduino Uno** es una placa de microcontrolador de código abierto basado en el microchip ATmega328P y desarrollado por Arduino.cc ,La placa está equipada con conjuntos de pines de E/S digitales y analógicas que pueden conectarse a varias placas de expansión y otros circuitos. La placa tiene 14 pines digitales, 6 pines analógicos y programables con el Arduino IDE (Entorno de desarrollo integrado) a través de un cable USB tipo B.

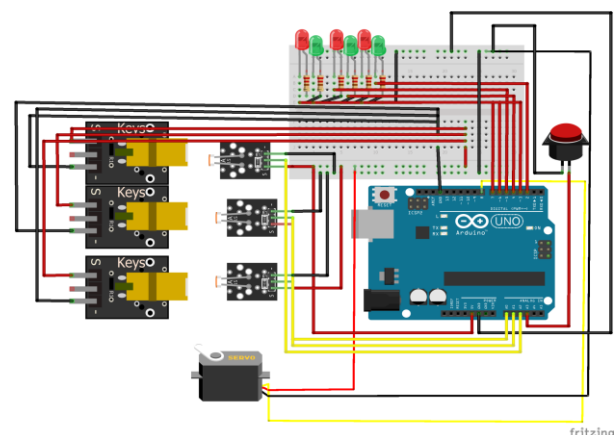
*Led:* Un diodo LED es un dispositivo que permite el paso de corriente en un solo sentido y que al ser polarizado emite un haz de luz.

*Resistencia:* La resistencia es una medida de la oposición al flujo de corriente en un circuito eléctrico

### Diseño general del sistema



### Diseño del montaje en fritzing



## Implementación del código

Inicialización de las variables globales en el ID ARDUINO

#include <Servo.h>: es la librería que nos permite controlar los giros del servo motor de una forma precisa en los grados que se requieran

Variables del botón: estas son asignadas a 5 variables de tipo INT

Variables de los leds : son las variables de salida digital para el control de los 6 leds que despone el sistema de los cuales con 3 rojos y 3 verdes indicando verde= libre y rojo=ocupado.

```
1  #include <Servo.h>
2  //VARIABLES DEL SERVO MOTOR
3  Servo  servo1;
4  int pinservo = 8;
5  int pmin = 1000;
6  int pmax = 2000;
7  //VARIABLES DEL BOTON
8  int botonviejo = 1;
9  int interruptor = A0;
10 int botonvalor = 0;
11 int aux = 3;
12 int botonnuevo;
13 //VARIABLES DE LOS LEDS
14 int ledverde1 = 2;
15 int ledrojo1 = 3;
16
17 int ledverde2=4;
18 int ledrojo2=5;
19
20 int ledverde3=6;
21 int ledrojo3=7;
```

Variables del fotorresistor: son las variables de entrada de la lectura analógica de las fotorresistencias

```
23 //VARIABLES DEL FOTORESISTOR
24 int pinentrada1 = A5;
25 int valorresistencia1;
26
27 int pinentrada2 = A4;
28 int valorresistencia2;
29
30 int pinentrada3 = A3;
31 int valorresistencia3;
32
33 int disponible;
34 int contador;
35 void setup() {
```

Void setup: son las declaraciones de las variables de cuáles serán entradas y salidas del sistema

Fotorresistor: se declaran como variables de entrada del sistema

Salidas de los leds: se declaran como salidas del sistema lo cual permite controlar cual del se en enciendo rojo o verde

Servomotor : es la variable encargada de controlar el giro del servomotor lo cual se le pasan 3 parámetros "pinservo, pmin, pmax"

Entrada del interruptor: es la variable encargada de recibir la señal enviada por el interruptor de cual permitirá el movimiento del servo que este requiera

```

35 void setup() {
36   Serial.begin(9600);
37   //FOTORESISTOR
38   pinMode(pinentrada1, INPUT);
39   pinMode(pinentrada2, INPUT);
40   pinMode(pinentrada3, INPUT);
41   //SALIDAS DE LOS LEDS
42   pinMode(ledverde1, OUTPUT);
43   pinMode(ledrojo1, OUTPUT);
44
45   pinMode(ledverde2, OUTPUT);
46   pinMode(ledrojo2, OUTPUT);
47
48   pinMode(ledverde3, OUTPUT);
49   pinMode(ledrojo3, OUTPUT);
50
51   //SERVOMOTOR
52   servo1.attach(pinservo, pmin, pmax);
53   //ENTRADA DEL INTERRUPTOR
54   pinMode(interruptor, INPUT);
55 }

```

Void loop: es el método que esta constantemente en un loop indeterminado hasta que el sistema se apague o tenga alguna falla

Entrada medición fotorresistor: se declaran las variables que van a recibir la señal de las resistencias de tipo analógicas

Interruptor para el servomotor: es un condicional que nos permitirá controlar la señal enviada por el interruptor

Donde **servo1.write(180)** hace el giro de 90 grados y **servo1.write(0)** vuelve al servo en el sentido contrario a su posición inicial

```

57 void loop() {
58   //ENTRADA MEDICION FOTORESISTOR
59   valorresistencia1 = analogRead(pinentrada1);
60   valorresistencia2 = analogRead(pinentrada2);
61   valorresistencia3 = analogRead(pinentrada3);
62
63   //INTERRUPTOR PARA EL SERVOMOTOR
64   botonuevo = digitalRead(interruptor);
65   delay(100);
66   if (valorresistencia1 <= 100 ||
67       valorresistencia2 <= 100 ||
68       valorresistencia3 <= 100)
69   {
70     if (botonuevo == 0 && botonviejo == 1)
71     {
72       if (botonvalor == 0) {
73         digitalWrite(aux, 1);
74         botonvalor = 1;
75         servo1.write(180);
76       }
77       else {
78         digitalWrite(aux, 0);
79         botonvalor = 0;
80         servo1.write(0);
81       }
82     }
83     botonviejo = botonuevo;
84   }

```

Lógica para el fotorresistor: son los condicionales que se nos permite encender y apagar cualquier led según lo que el sistema requiera donde la condición es si el valor es menor o igual 100 realiza la acción indicada

**Valorresistencia:** es la señal recibida por parte de las fotorresistencias

**LOW:** indica apagar led

**HIGH:** indica prender led

```

97   //LOGICA PARA EL FOTORESISTEOR
98   // Y LOS LEDS
99   if (valorresistencia1 <= 100) {
100     digitalWrite(ledverde1, LOW);
101     digitalWrite(ledrojo1, HIGH);
102
103   } else {
104     digitalWrite(ledverde1, HIGH);
105     digitalWrite(ledrojo1, LOW);
106   }
107   if (valorresistencia2 <= 100) {
108     digitalWrite(ledverde2, LOW);
109     digitalWrite(ledrojo2, HIGH);
110

```

## Monitoreo de disponibilidad

Se suma el total de los valores de las 3 fotorresistencias para poder contar cuantas plazas disponibles se tienen en el momento para imprimir el valor de disponibilidad que puedes ser de 0 plazas a 3 plazas como máximo

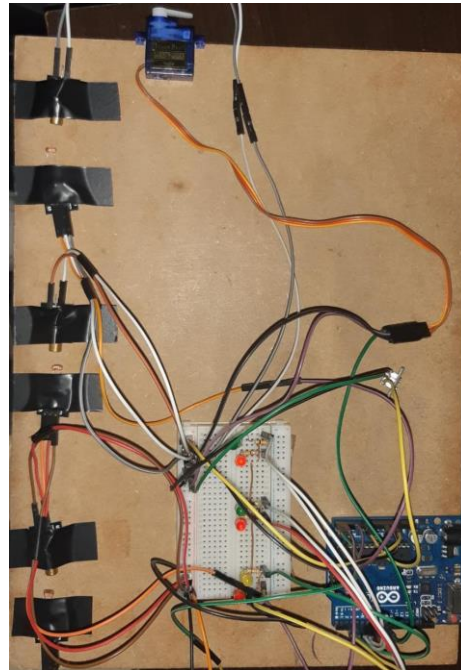
```
124 //monitoreo de disponibilidad
125 disponible = valorresistencia1
126           + valorresistencia2
127           + valorresistencia3;
128
129 if (disponible <= 300) {
130     contador = 3;
131 }
132
133 if (disponible >= 400) {
134     contador = 2;
135 }
136
137 if (disponible >= 950 && contador == 2) {
138     contador = 1;
139 }
140
141 if (disponible >= 2000 && contador == 1) {
142     contador = 0;
143 }
144
145 Serial.println(contador);
146 }
```

## Conclusiones

Con el desarrollo e implementación de un Sistema de Parqueadero Inteligente, se pudo comprobar que es factible diseñar un sistema que gestione un parqueadero. En base a la investigación realizada y a la información obtenida de libros, artículos, se determinó los equipos y tecnología necesarios para la elaboración de un Sistema de Parqueo Inteligente tales como: Tarjeta Arduino uno y los diferentes sensores y actuadores.

Cabe resaltar que este parqueadero tiene algunas limitaciones las cuales podrán ser corregidas en un futuro como lo puede ser que utilice tecnología IoT , para una mejor visión y control de este

## Implementación del parqueadero



## Referencias bibliográficas

- [1] Club Ensayos 2012.  
<http://clubensayos.com/Historia/TipoDe-Estacionamientos/161627.html> (último acceso: 5 de Marzo de 2012)
- [2] Parking Management International, 2009. Puzzle Parking System. Parking Management International. Macedonia.  
[http://www.pmieu.com/products\\_puzzle\\_en.html](http://www.pmieu.com/products_puzzle_en.html) (último acceso: 5 de Marzo de 2012).
- [3] Sanders, Shannon 2010. Aparcamiento automático ahorra espacio en lugares estrechos. Intransition Magazine. New Jersey.  
[http://www.intransitionmag.org/SpringSummer\\_2010/automated\\_parking.aspx](http://www.intransitionmag.org/SpringSummer_2010/automated_parking.aspx) (último acceso: 7 de Marzo de 2012).
- [4] Suvidha Parking Lift. 2009. Puzzle Parking System. Suvidha Parking Lift. Mumbai, India.  
<http://www.suvidhaparklift.net/puzzle-parking-system.htm> (último acceso: 8 de Marzo de 2012). 258
- [5] Suvidha Parking Lift. 2009. Puzzle Parking System. Mumbai, India. <http://www.suvidhaparklift.net/puzzle->

parking-system.htm (último acceso: 8 de Marzo de 2012).

[6] Parking Management International, nd. Puzzle Parking System. Parking Management International. Macedonia.

[http://www.pmieu.com/products\\_puzzle\\_en.html](http://www.pmieu.com/products_puzzle_en.html)  
(último acceso: 5 de Marzo de [1])