

Universidad Nacional de La Matanza

Departamento de Ingeniería e Investigaciones Tecnológicas

Sistemas Operativos Avanzados

Trabajo Práctico

Sistemas Embebidos Android

Estacionamiento

|  |  |
| --- | --- |
| Apellido y Nombre | DNI |
| Devanna, Matias | 35.202.202 |
| Monzón, Nicolas | 36.635.596 |
| Giupponi, Brian | 37.109.479 |

Objetivo:

Realizar un sistema de estacionamiento de vehículos, que permita facilitar todo el proceso que conlleva estacionar, desde que el usuario considera entrar al estacionamiento hasta el correcto estacionamiento de su vehículo. Este sistema tendrá un sistema embebido compuesto por un módulo bluetooth para controlar la reserva de una plaza de estacionamiento, adicionalmente la placa se conectará a un puente H para controlar el motor de una barrera que permita ingreso/salida al estacionamiento controlado desde un sensor infrarrojo. También se hará uso de sensores de ultrasonido para orientar al usuario al estacionar y un sensor de fotorresistencia que controlará la luminosidad en la plaza de estacionamiento.

Hardware utilizado:

* 1 Placa Arduino UNO
* 1 Modulo bluetooth HC-05(Maestro-Esclavo)
* 1 Puente H
* 6 LED’s(2 Rojos, 2 Blancos y 2 Verdes)
* 1 Protoboard 400 puntos
* 2 Batería 9V
* 2 Interruptores De Palanca 3 Pines Na/nc 3a/250v Mona
* 2 Sensores ultrasonido - Hc-sr04
* 1 Motor de traba puertas eléctrico
* Cables macho – macho/ Macho-Hembra
* 2 Sensores infrarrojo
* 1 Sensor de fotorresistencia
* 2 Resistencias 4k7

Software utilizado:

* Android studio 2.3
* Arduino 1.8.2

Links utilizados:

* <https://github.com/patriotaSJ/Bluetooth> (conexión al bluetooth).
* <http://www.maestrosdelweb.com/curso-android-sensores-trabajar-con-acelerometro/> (utilización de acelerómetro en Android)
* <https://medium.com/@victor.garibayy/obteniendo-mi-ubicaci%C3%B3n-en-android-studio-377226910823> (Sensor GPS)
* <https://www.luisllamas.es/detectar-obstaculos-con-sensor-infrarrojo-y-arduino/> (Sensor Infrarrojo).
* <http://www.geekfactory.mx/tutoriales/tutoriales-arduino/sensor-ultrasonico-hc-sr04-y-arduino/> (Sensor ultrasonido).

Código:

Tareas realizadas:

1. Realizamos la conexión al módulo de bluetooth para realizar el intercambio de mensajes entre la aplicación y la placa Arduino. La distancia máxima de medición de este módulo es de 10m.
2. Conectamos el sensor infrarrojo para detectar vehículos en la entrada al estacionamiento. Este sensor está configurado para detectar vehículos a una distancia de 4cm.
3. Conectamos la placa del puente H para poder controlar el movimiento de la barrera del estacionamiento.
4. Conectamos el sensor ultrasonido para medir la distancia del vehículo a la pared de la plaza. Este sensor está configurado para medir hasta una distancia de 18cm.
5. Conectamos un led blanco (que trabaja en conjunto con el sensor ultrasonido) en cada plaza del estacionamiento que a mayor frecuencia de parpadeo indicará que el vehículo se encuentra más cerca de la pared. Este led parpadea cuando el vehículo se encuentra a una distancia de la pared entre 6cm y 18 cm.
6. Conectamos un sensor de fotorresistencia para medir la luminosidad del estacionamiento en base a la cantidad de luz en el ambiente. Se configuró que si el valor de medición es mayor 100 consideramos que está oscuro el ambiente por lo cuál seteamos una intensidad de 100 de los leds(como máximo pueden tener una intensidad de 250), si el valor no es mayor a 100 seteamos una intensidad de 200 de los leds.

Información técnica:

Sensores ultrasonido - Hc-sr04:



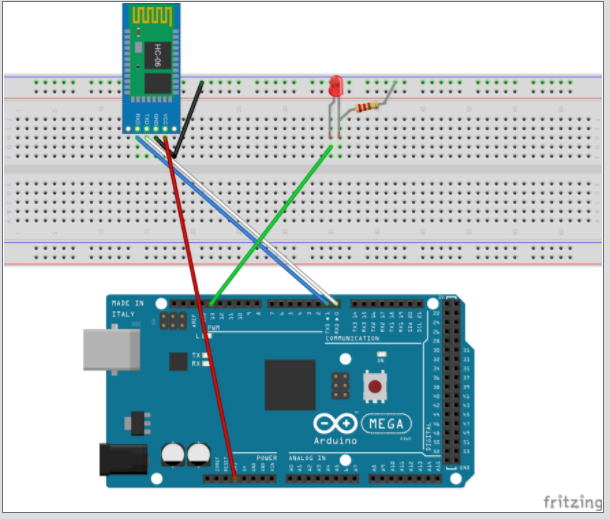
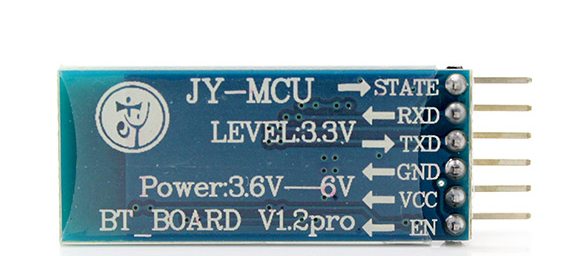
### Pines de conexión:

* VCC
* Trig (*Disparo del ultrasonido*)
* Echo (*Recepción del ultrasonido*)
* GND

## Características

* *Dimensiones del circuito: 43 x 20 x 17 mm*
* *Tensión de alimentación: 5 Vcc*
* *Frecuencia de trabajo: 40 KHz*
* *Rango máximo: 4.5 m*
* *Rango mínimo: 3 cm*
* *Duración mínima del pulso de disparo (nivel TTL): 10 μS.*
* *Duración del pulso eco de salida (nivel TTL): 100-25000 μS.*
* *Tiempo mínimo de espera entre una medida y el inicio de otra 20 mS.*

Conexión Modulo Bluethood HC-05:

El módulo de bluetooth HC-05 dispone de 6 pines y puede actuar como maestro-esclavo. En la imagen anterior se muestra la conexión básica para el funcionamiento del módulo.

Características del módulo Bluetooth HC05:

- Fácil de usar y completamente encapsulado  
- Chipset: CSR  
- Bluetooth V2.0  
- Alimentación: 3.6V a 6V  
- Niveles lógicos: 3.3V  
- Velocidad de transmisión: 1200, 2400, 4800, 9600, 19200, 38400, 57600 y 115200 . Seteable por comandos AT  
- Configuración predeterminada del puerto serie: 38400, N, 8, 1



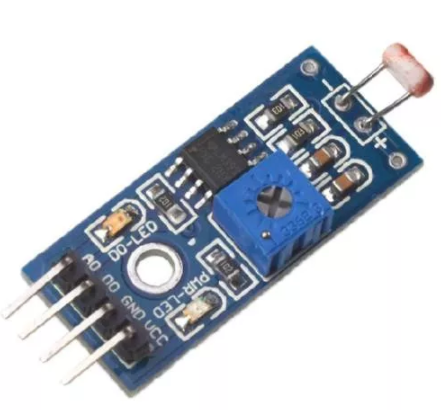
Conexión con puente H L298:

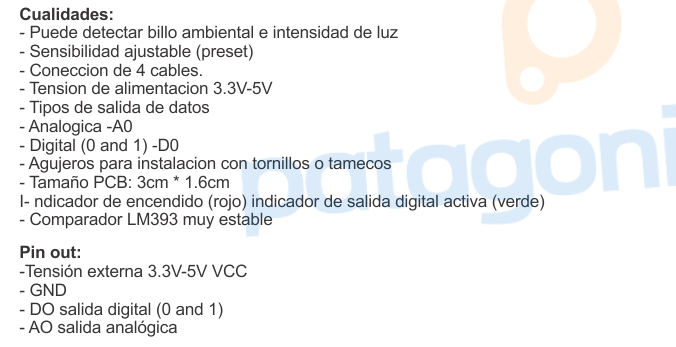
 

Este módulo basado en el chip L298N te permite controlar dos motores de corriente continua o un motor paso a paso bipolar de hasta 2 amperios. El módulo cuenta con todos los componentes necesarios para funcionar sin necesidad de elementos adicionales, entre ellos diodos de protección y un regulador LM7805 que suministra 5V a la parte lógica del integrado L298N. Cuenta con jumpers de selección para habilitar cada una de las salidas del módulo (A y B). La salida A está conformada por OUT1 y OUT2 y la salida B por OUT3 y OUT4.

Resistencias 4k7:

Sensor fotorresistencia:





Baterías 9v:



Características:

Tensión nominal: 9V   
Tensión de operación: 9,6 – 4,8 V   
Capacidad: 565 mAh   
Peso: 45 gramos   
Volumen: 22,8 cm3   
Temperatura de operación: 20ºC a 54ºC   
Conector: PP3   
Denominación ANSI: 1604A   
Denominación IEC: 6LR61

Interruptores de palanca 3 pines na/nc 3a/250v Mona



Características:

interruptor de palanca 123 MTS-3 pines spdt de encendido / apagado / encendido de 3 posiciones momentáneo

caracteristicas:

-Environmental protección resistente a altas temperaturas

-con el tercer grado puede ser ajustable

-single polo doble tiro, situ (en)

protección -Environmental resistencia a altas temperaturas

presupuesto:

5a 120 VCA, 250 Vac 2a

resistencia a controlar: 20 mW máx

resistencia de aislamiento: 500 V CC min 1000mω

temperatura de funcionamiento: -25 ℃ ~ + 85 ℃

esperanza de vida: 10000times

función: MTS-123 situ (en)

agujero de instalación: 6mm

Material: Plástico + cobre

Leds alta luminosidad:

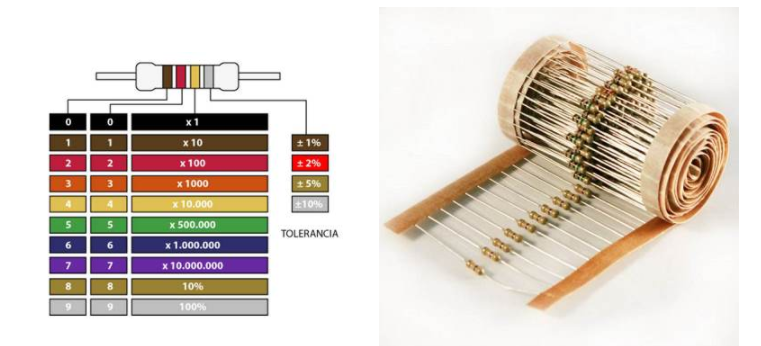


Voltaje: 3.0 ~ 3.2V

Corriente: 20MA

Longitud de onda: 11000-13000

Resistencias 4k7:

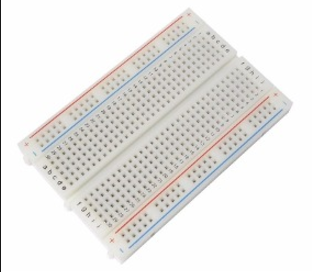


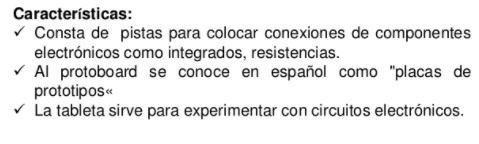
Resistencias para controlar la cantidad de corriente. Se oponen al paso de la corriente, en nuestro caso utilizamos las de 4k7 en los switchs.

Motor traba puerta:



Protoboard 400 puntos:

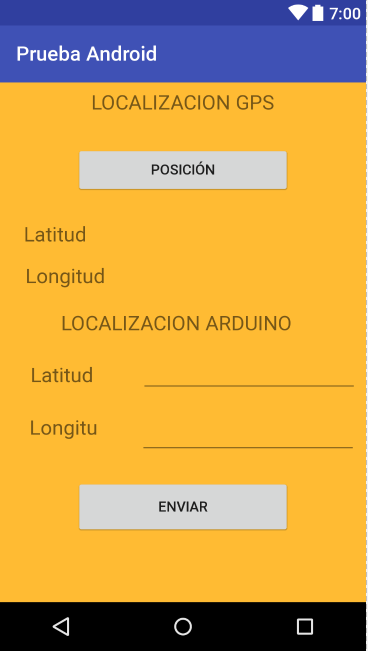




* 2 Sensores infrarrojo
* 1 Sensor de fotorresistencia

Armado de la aplicación Android para la comunicación arduino:

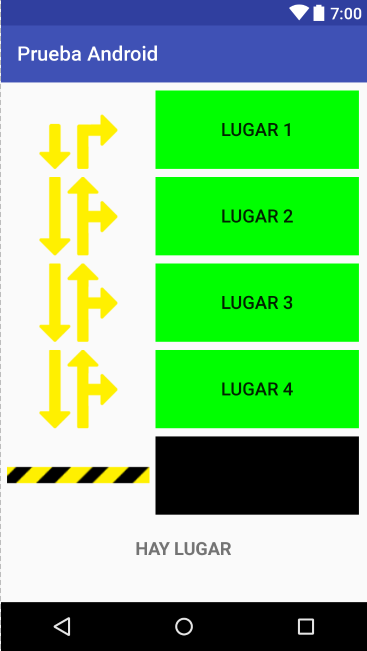
**Activity: Activity\_main** 🡪 En este activity se toma la posición del dispositivo Android y se toma una posición ficticia del lugar geográfico del estacionamiento. Si la distancia entre los 2 puntos es de hasta 1km se permite ingresar en la sección de reservas.



Posición: Devuelve la localización geográfica del dispositivo.

Enviar: Toma los datos de la posición del dispositivo más la latitud y longitud en donde nosotros establecemos de dónde se encuentra del estacionamiento, calcula la distancia y si está alrededor de 1km, permite dirigirse a la activity de reserva de la playa.

**Activity: activity\_elegir\_lugar** 🡪 A través de este activity es posible para el usuario que pueda visualizar las plazas del estacionamiento que se encuentran libre, y aquellas que están ocupadas. Si alguna se encuentra libre será posible realizar una reserva de la misma. Por default se ha definido que la reserva será de 10 segundos luego de consumido ese tiempo la plaza volverá a estar libre para que pueda ser ocupada o bien ser reservada por otro conductor. Si la plaza está reservada y el conductor llega al estacionamiento, sacudir el dispositivo provocará que el led verde de la plaza reservada comience a parpadear para que el conductor sepa dónde debe estacionar.



Estados de las plazas de estacionamiento:

* Led de plaza color verde: Plaza disponible para ser ocupada o reservada por un conductor:
* Led de plaza color rojo titilando: Plaza reservada.
* Led de plaza color verde titilando: Indica qué plaza fue reservada por el usuario.
* Led de plaza color rojo: Plaza ocupada.

Estados de las plazas de la aplicación:

* Lugar color verde: Libre
* Lugar color azul: Reservado
* Lugar color rojo: Ocupado

Información adicional de la maqueta:

Utilizamos para la maqueta cartón de 1m x 70cm, el cuál fuimos cortando con un cutter para ir dándole forma al estacionamiento. Fuimos pegando cada parte con una pistola en colador, por otra parte encintamos los cables y los pegamos a las paredes para una mayor prolijidad.

* Imágenes de la maqueta:

