

# Diseño, desarrollo y construcción de robot explorador controlado a distancia.

Zambrano Alejandro y Calderón Ángel  
{2420152006, 2420152009}@estudiantesunibague.edu.co  
Universidad de Ibagué

**Abstract**—It is intended to design and build an explorer robot, with the possibility of controlling it remotely by the Internet and observing the route by a camera in it. Two main boards are used for this project: The first one, a Raspberry Pi, based on the SSH communication protocol and processing capacity, is used to analyze the delivered data and take the main decisions that the driver bring to it. The second one, An Arduino to control all the analogous elements and to take fast measures. The result obtained is a vehicle with lighting system, distance measurement and real time video to achieve an effective management.

## I. INTRODUCCION

En la actualidad existen diferentes prototipos de ‘robots’ (Sumo, seguidores de línea, resuelve laberintos, exploradores, etc.) con respectivas cualidades que identifican a cada uno de ellos. La práctica que se desarrollara pretende mostrar paso a paso la construcción de un robot explorador, el cual está en capacidad de ser maniobrado desde un lugar diferente donde éste se encuentre. El objetivo de este proyecto es poder avanzar por terreno desconocido, investigar y resolver diferentes obstáculos que se presenten durante el camino.

Sobre este tipo de robot se ha trabajado en diferentes aspectos y complejidades. En la NASA se ha usado esta topología para explorar diferentes lugares (como la Luna y Marte), logrando una manipulación del mismo y así poder descubrir más aspectos importantes que aportan a investigaciones sobre el comportamiento del espacio. Es también muy usado por las fuerzas armadas de diferentes países en terrenos hostiles para desactivar artefactos explosivos e investigar posiciones estratégicas del enemigo; gracias a esta tecnología se pueden evitar poner en riesgo más vidas humanas.

Por ultimo pero no menos importante, son los concursos a nivel educativo donde se realizan equipos con el fin de resolver una respectiva pista, esquivar obstáculos y realizar tareas específicas durante el trayecto. El concurso se ha venido desarrollando en Stillwater Oklahoma, Estados Unidos, donde el robot debe ser operado por lo menos a 80 kilómetros de distancia y atravesar por una pista la cual desconoce el piloto. De igual manera se ha implementado en Colombia en conjunto a la Corporación Unificada Nacional de Educación Superior (CUN). Este tipo de trabajo busca incentivar el desarrollo en la rama de la tecnología, creatividad y diseño [1].

Bajo las características del concurso nombrado anteriormente, se basa el diseño del proyecto. El piloto observara el terreno por donde circulará mediante una cámara articulada en el chasis del vehículo. El robot también incorpora: un sistema de iluminación basado en LED’s, en caso de que el entorno carezca de claridad. Sensores de ultrasonido de distancia para poder conocer de manera más

asertiva cuan alejado se encuentra y poder avanzar de manera efectiva, evitando choques.

Se dio uso de dos tarjetas: Arduino y RaspberryPi. La primera es la encargada del correcto funcionamiento de los sensores de distancia; de proporcionar un efectivo voltaje a los motores mediante la modulación por ancho de pulso (PWM) y supervisar el nivel de la batería a implementar. La segunda, programada en python, establece la conexión con el piloto; enciende y/o apaga el sistema de iluminación; toma decisiones respecto a los motores como su activación y dirección.

## II. DESARROLLO

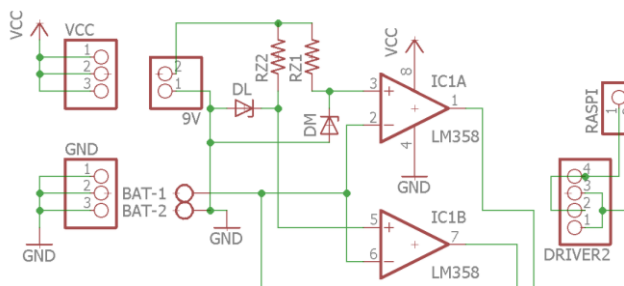
Se tomaron varios aspectos para el diseño del robot explorador. Los primeros dan la elección de las dos tarjetas principales. Se da uso de una Raspberry Pi 2 Modelo B (RPI), la cual establece la correcta comunicación entre el piloto y la máquina mediante un adaptador wifi USB ‘plug and play’; genera la interfaz de control con el usuario; procesa la información ingresada por teclado y realiza la toma de decisiones dependiendo del diseño. Luego, se tuvo en cuenta las partes análogas requeridas como el control del motor y los sensores de distancia, llevando acabo la implementación de un Arduino Mini-Pro y su respectivo modulo conversor USB a TTL para garantizar una comunicación serial con la RPI. Ambas tarjetas se alimentaran con una power-bank usada normalmente para cargar celulares.

Una vez establecido lo anterior, se deciden los motores a usar realizando una aproximación al peso el cual cargara. Con esto y una guía del ingeniero guía, se eligen motores con caja reductora de relación 34:1, 6.5kg-cm, 280RPM. Determinado estos, se procede con la alimentación. Con ayuda de una fuente de laboratorio se hacen pruebas con los motores para verificar cuál será su consumo aproximado y basado en ello se elige una batería de litio polímero de dos celdas (7.4v) 2200mAh con una descarga de 30C continuo.

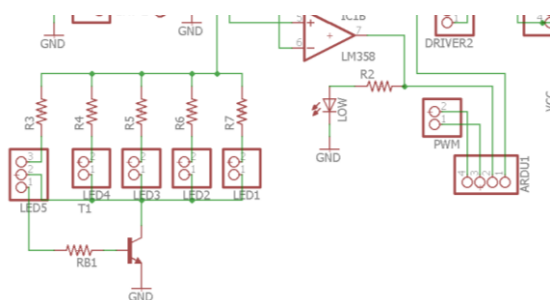
De la misma prueba anterior con los motores se elige el driver que controle los mismos. El driver L290N ya que proporciona dos canales y una corriente máxima de 2.5A por cada uno, con control digital sobre el giro de los motores y otro pin aparte para el PWM de los mismos. Para el sistema de iluminación, se eligieron nueve LED’s de 10mm difusos. Como estos usualmente son de 3v a 20mA y la batería es de 7.4v, se realizan cuatro series de dos LED’s y el restante se conecta solo. Finalmente dentro de los componentes electrónicos se elige como sensor de distancia el HC-SR04 que provee un alcance máximo de 450cm con un ángulo no mayor a los 15°.

Terminando la parte mecánica, se eligen llantas con un diámetro de 120mm con 60mm de grosor con un neumático de caucho negro y picos para mayor tracción. Con base a las dimensiones de las llantas se elige la rueda loca. Por problemas de disponibilidad se elige una de 3cm de alto y se le logra dar más altura con unos separadores plásticos, como se verá más adelante. Dentro de los implementos extra, se encuentran: tornillos, con su respectiva tuerca, de 3x60mm y de 5x200mm; caimanes con cable grueso para soportar la corriente de arranque; cables tipo 'jumper' para conexiones generales entre las dos placas y una 'auxiliar' nombrada a continuación.

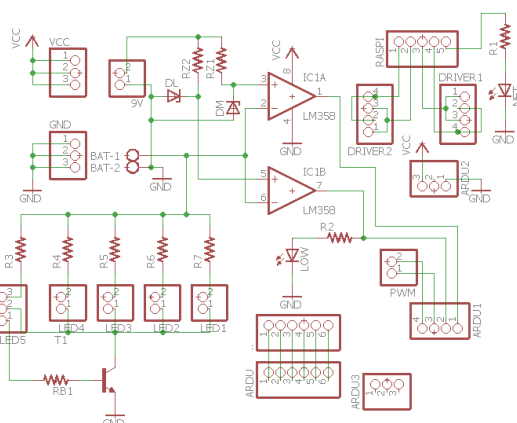
Con el fin de hacer un diseño más liviano se diseña una placa 'auxiliar' cuyo esquemático se presenta a continuación:



Parte uno – Placa auxiliar



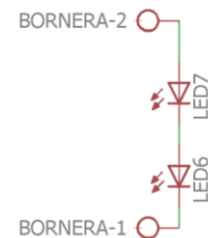
Parte dos – Placa auxiliar



Circuito de la placa auxiliar

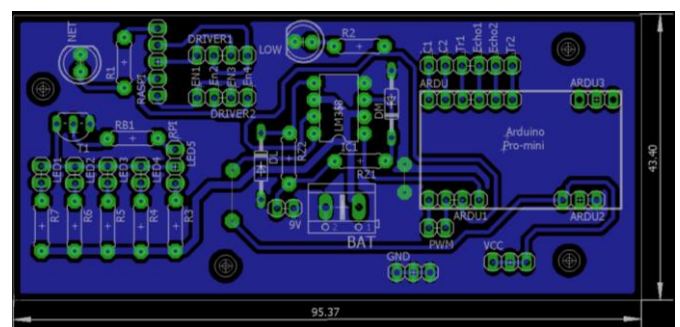
El circuito básicamente incluye las borneras para conectar el driver, los LED's de iluminación, el arduino, los pines de control de la RPI y la alimentación de los sensores. Los dos operacionales trabajan como comparadores entre un voltaje en su entrada no inversora con el voltaje de la batería.

Los voltajes a comparar se proporcionan por diodos zener. Su valor dependerá del voltaje de batería que se halla elegido y el nivel que se requiera proteger. RZ1 y RZ2 se calculan para garantizar una correcta polarización de DM y DL. Los conectores LED 1, 2, 3, 4 y 5 van a los LED's de iluminación de la siguiente manera cada uno:



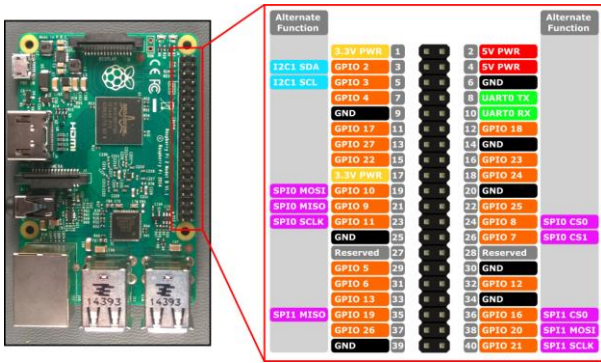
Serie de LED's para sistema de iluminación

Los demas pines nombrados 'ARDU' son los pines donde ira puesto el arduino elgido. Para garantizar que los pines encagen correctamente con la placa y sus dimensiones, se puede ingresar a la pagina oficial de arduino ([www.arduino.cc](http://www.arduino.cc)) buscar por el tipo de placa deseado y descargar los archivos respectivos que contienen el diseño de la misma. Acontinucion se expone el PCB que se observara mas adelante en el montaje (las dimensiones del mismo se encuentran en mm):



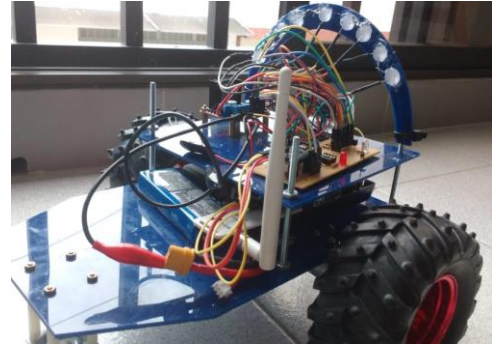
Los LED's de 3mm 'NET' y 'LOW' indican las desconexión con el piloto y batería baja, respectivamente. Para el transistor que prendera o apagara los LED's de 10mm se usa uno que logre soportar minimo 150mA y que tenga una hfe alta para una mejor saturacion (se uso el 2N2222A); las resistencias R3, 4, 5, 6 y 7 se calculan por ley de ohm donde el voltaje sobre ellas sera la diferencia de potencial entre la batería (en nuestro caso 7,4) y el consumo de los LED's (6v para la configuracion en serie y 3v para el LED independiente) y la corriente de la misma sera de 20mA. La bornera '9V' se diseña para conectar una pila cuadrada de 9v con el fin de mantener el menor cambio en la referencia de los comparadores.

Los pines en el código de la RPI se deben tener en cuenta con su equivalencia en la tabla de GPIO. A continuación se observan los equivalentes de los pines físicos y su respectiva característica de GPIO. En la sección de resultado se exponen los links respectivos, información adicional y los códigos relacionados.



Pines GPIO - Raspberry Pi

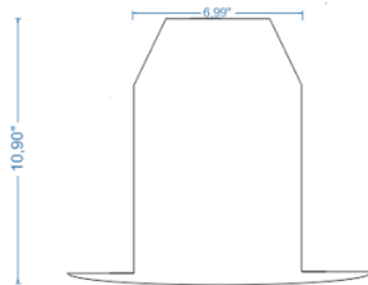
### Montaje final del robot



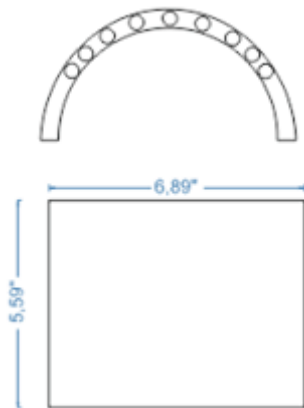
Vista lateral derecha

## III. RESULTADOS

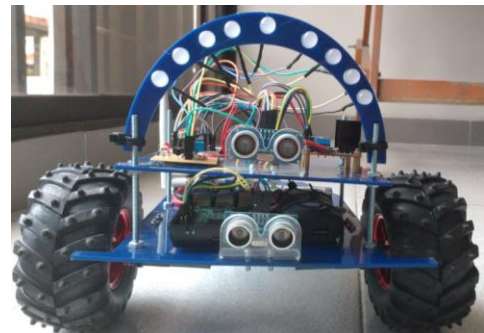
### Chasis del Robot



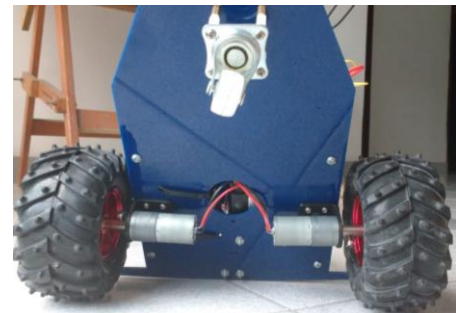
Primer piso



Segundo piso y barra de LED's



Vista frontal

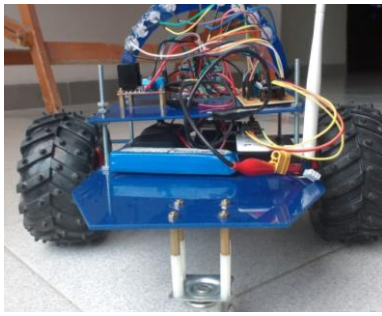


Vista inferior

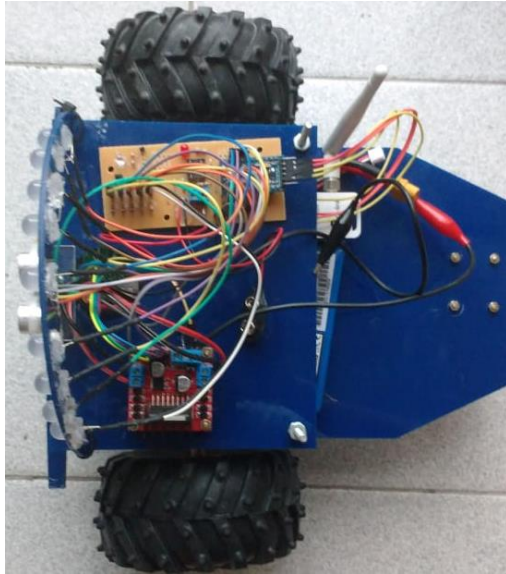


Vista lateral izquierda

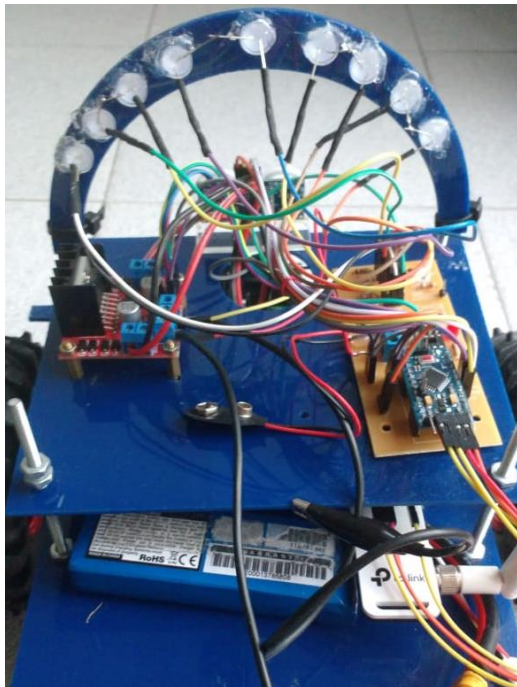




*Vista trasera*



*Vista superior*



*Vista ampliada*

- **Link del funcionamiento del robot:**

<https://www.youtube.com/watch?v=lbbPQjSrndw>

- **Link de los codigos relacionados:**

<https://github.com/utopia2000/Blue-Bot->

#### IV. CONCLUSIONES

El tipo de comunicación no fue del todo efectivo ya que la Raspberry pi 2 mantiene unas irregularidad debido a que no incluye un módulo WIFI, por ende el uso de una antena alterna a largas distancias provoca constante desconexión y lentitud de la misma.

Las ruedas a utilizar se deben compensar con el tamaño y peso del chasis para que se disponga de mejor manejo, ya que la inestabilidad del robot al frenar y hacer maniobras bruscas provocan un riesgo contundente sobre el hardware.

Se debe garantizar una buena calidad de los motores y que las características de ambos sea iguales, para evitar problemas de fuerza y lograr que se desplace de forma recta. Además, éstos no deben presentar mucha resistencia al intentar arrancar, ya que se vuelve una carga más y por ende se exige mayor corriente pudiendo exceder los límites del driver elegido e impidiendo que, por ejemplo, en una subida pendiente logre moverse.

Los fotogramas por segundo (fps) son indispensables para la cámara de uso, ya que de esto, dependerá si el piloto se encuentra viendo en tiempo real o no, el movimiento del robot; además se debe atender a un balance entre calidad del video y velocidad del mismo, ya que a mayor resolución más lenta puede ser la comunicación. En general, si es posible, es mejor ofrecer una comunicación por canal alterno para conseguir una mejor disposición y un pequeño tiempo de toma de decisiones.

Si se desea usar dos motores con dos llantas y una rueda loca o cuatro llantas con dos motores, es recomendable balancear el peso sobre los motores para evitar más trabajo de ellos y si se es posible usar cuatro llantas con cuatro motores el peso es mejor balancearlo en la mitad del robot.

Uno de los puntos a tomar en cuenta de la Raspberry, es que no tiene la capacidad de tomar lecturas análogas no, lo que justifica una ayuda externa como lo es el Arduino, manteniendo una conexión entre estos dos elementos. Por ello si es posible usar sensores digitales o lecturas procesadas digitalmente, de esta manera se reduce los costos de construcción del mismo.

#### REFERENCES

- [1] OSU Mercury Robotics. *Mercury challenge*. Disponible en: <https://mercury.okstate.edu/content/mercury-challenge>
- [2] R. Matt y W. Shawn. "Getting started with raspberry pi". 3rd ed., San Francisco C.A.: Maker Media 2016.
- [3] Python Software Foundation [US]. *Python 3.6.5 documentation*. Disponible en: <https://docs.python.org/3/>

