

# Introducción

Este manual tiene el propósito de informar acerca del armado y descripción del robot educativo que estaremos utilizando en la competencia de futuros ingenieros.

Como participantes de la competencia y quienes diseñamos el ejemplar, queremos cumplir con el objetivo de desarrollar una programación y armado sencillo del cual podamos acceder a conocimientos de robótica, especialmente con Spike.

Mediante este documento proporcionamos la metodología detallada que utilizamos para realizar nuestro proyecto.

Para la categoría de futuros ingenieros nuestro equipo Starlights (Modelo con LEGO)

# Materiales

- Chassis de kit de lego
- sensor ultrasónico
- Hub
- 2 motores grandes
- Sensor de color
- Dos ruedas grandes
- Dos ruedas medianas

# Instrucciones de montaje

Entrevistando a nuestro ingeniero nos redacto el como realizo el armado del modelo, Se utilizaron dos motores grandes como base, también esta inspirado en un auto móvil de formula 1, solo que se desenvuelve creativamente un poco distinto al tener el cambio de dos ruedas grandes como tracción trasera y acortar su tamaño con el modelo característico, el Hub el cual es el corazón del proyecto se encuentra en un lugar visible para poder encender y apagarle cuando es necesario, las dos ruedas delanteras se encargan de la dirección, se colocaron por ultimo el sensor de color y ultrasónico.

# Introducción a Spike

Experiencia de aprendizaje STEAM, con la construcción y programación de modelos, se puede construir de manera creativa, siguiendo manuales y desafíos para el desarrollo de los estudiantes, con diversas piezas como sensores, motores y demás para la libre expresión de quien lo utilice, aprenden a solucionar de manera crítica, ideal no solo para clases y actividades extracurriculares si no también en casa.

# Programación del robot (Python)

```
from spike import PrimeHub, Motor, MotorPair, DistanceSensor, ColorSensor
```

```
from spike.control import wait_for_seconds, wait_until
```

```
from spike.operator import less_than
```

```
# Inicialización de hub y sensores/motores
```

```
hub = PrimeHub()
```

```
motor_trasero = Motor('A')# Motor de tracción trasero en el puerto A
```

```
motor_direccion = Motor('B')# Motor de dirección en el puerto B
```

```
sensor_distancia = DistanceSensor('D')# Sensor de distancia en el puerto E
```

```
sensor_color = ColorSensor('F')# Sensor de color en el puerto F
```

```
# Variables de control
```

```
contador_lineas_azules = 0
```

```
# Configuración inicial de motores
```

```
motor_trasero.set_default_speed(100)# Ajusta la velocidad al valor original
```

```
motor_direccion.set_default_speed(100)# Ajusta la velocidad al valor original
```

```
# Función para mover el robot hacia adelante
```

```
def mover_adelante():
```

```
    motor_trasero.start()
```

```
# Función para detener el robot
```

```
def detener():
```

```
    motor_trasero.stop()
```

```
# Función para girar el robot a la derecha
```

```
def girar_derecha():
```

```
    motor_direccion.run_for_degrees(90)# Ajusta el valor según sea necesario
```

```
# Función para girar el robot a la izquierda
```

```
def girar_izquierda():
```

```
    motor_direccion.run_for_degrees(-90)# Ajusta el valor según sea necesario
```

```
# Loop principal
```

```
while True:
```

```
    distance = sensor_distancia.get_distance_cm()
```

```
    color = sensor_color.get_color()
```

```
# Verificar detección de líneas azules
```

```
if color == 'blue':
```

```
    contador_lineas_azules += 1
```

```
    if contador_lineas_azules >= 12:
```

```
        detener()
```

```
        wait_for_seconds(3)
```

break# Termina el loop principal para detener el robot después de 3 segundos

# Verificar detección de líneas rojas para girar a la derecha

if color == 'red':

    girar\_derecha()

    wait\_for\_seconds(0.5)

# Verificar detección de obstáculos para esquivar por la izquierda

if distance is not None and distance < 20:# Si hay un obstáculo a menos de 20 cm

    detener()

    wait\_for\_seconds(0.5)

    girar\_izquierda()

    wait\_for\_seconds(0.5)

    mover\_adelante()

    wait\_for\_seconds(0.5)

    girar\_derecha()

    wait\_for\_seconds(0.5)

    detener()

    wait\_for\_seconds(0.5)

else:

    mover\_adelante()

```
wait_for_seconds(0.1)
```



## Guía de resolución de problemas

Al ser un proyecto considerablemente sencillo no hubo problemas en su desarrollo a excepción de la cantidad que se utilizan de conectores pues los de un solo kit no suelen ser suficientes, y el tener que utilizar únicamente 2 motores fue un poco mas limitante pero se encontró una solución coherente utilizando ambos lados por separado de un mismo motor grande.

## Referencias y recursos

LEGO (2020). Recuperado de: [LEGO® Education SPIKE™ Prime Set 45678 | LEGO® Education |](#)

[Buy online at the Official LEGO® Shop US](#)

Mohd Shahid (28 de febrero de 2020). Recuperado de: [Obstacle Avoiding Robot 4 wheel |](#)

[Obstacle avoiding car | Techatronic](#)

## Anexo



