



UNIVERSIDAD NACIONAL DE LA PLATA

Facultad de Informática - UNLP
Cloud Computing y Cloud Robotics (00A22)
Curso 2022 - Trabajo Final

Grupo 10 y 14

Alumnos:

Stella Joaquín, Guerrero Nicolás, Madonia Valentina, Demoro Mauricio

Índice

1 - Presentación de la problemática a resolver	3
2 - Elementos utilizados	3
2.1 - Hardware	3
2.1.1 Conexionado de elementos	7
2.2 Software	9
3 - Demostración del funcionamiento	11
Fuentes	11

1 - Presentación de la problemática a resolver

La problemática consistía en realizar el control remoto de un vehículo no tripulado controlado por una computadora Raspberry Pi, haciendo uso de la interfaz de E/S del mismo y distintos sensores, motores y servos.

El funcionamiento de nuestra solución es la siguiente: nuestro vehículo consistía de un motor que controlaba las dos ruedas traseras, un servo para controlar el giro de las ruedas delanteras, un puente H para controlar la dirección del motor, un sensor de ultrasonido ubicado en la parte delantera y un tira de LEDs de color verde, amarillo y rojo.

Desarrollamos una interfaz para dispositivos móviles en la se puede controlar la dirección de los motores y la velocidad de avance/retroceso y encender y apagar los motores. Si el sensor detecta un objeto a menos de 35 centímetros por delante, frenará y no permitirá avanzar más, solo retroceder. Además, el LED verde indica cuando está avanzando y su intensidad es relativa a la velocidad del motor. Así mismo sucede con el LED amarillo pero cuando retrocede. Y por último, el LED rojo indica si se bloqueó el avance por tener un objeto cerca.

2 - Elementos utilizados

2.1 - Hardware

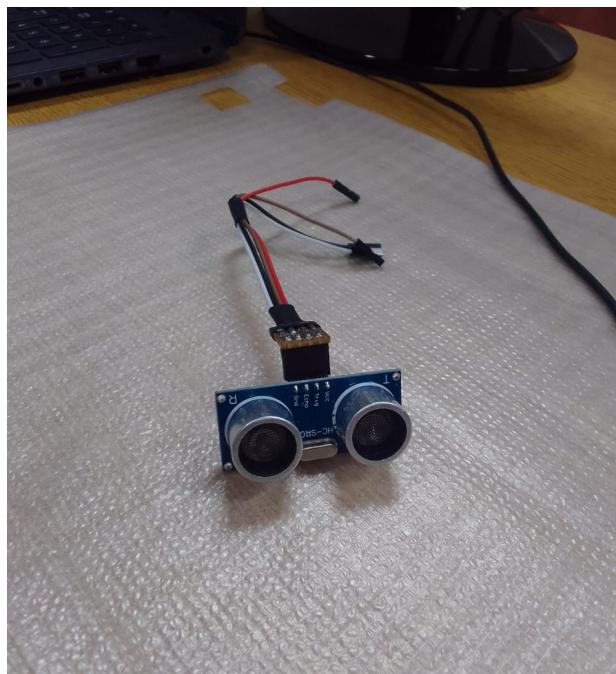
- Raspberry PI 3 Model B v1.2 (2015)

Es una computadora de placa única o single board computer (SBC) de bajo costo desarrollada por la Raspberry Pi Foundation en asociación con Broadcom. Inicialmente desarrollada con fines educativos, su uso se extendió a otros campos como la robótica, IoT, domótica y monitoreo en distintas aplicaciones. Dispone de un procesador ARMv8, 4 puertos USB, conexión HDMI, Ethernet, Wifi, Bluetooth, un slot para una tarjeta de memoria Micro SD, y 40 pines GPIO digitales (General Purpose In Out), que permiten recibir información de sensores y enviar señales a actuadores.



- Sensor de ultrasonido HC-SR04

Es un sensor de distancia de bajo costo que utiliza ultrasonido para detectar la distancia de un objeto en un rango de 2 cm a 450 cm. Dispone de 5 pines: uno para recibir una alimentación de 5V; un pin Ground, en tierra; un Trigger Pulse Input, que permite recibir una señal que dispare la detección de aproximación; y un Echo Pulse Output, que retorna la distancia. Para calcular la distancia transmite una señal de 40kHz y, tras rebotar esta en un objeto, la recibe por un receptor. Tras esto, calcula la distancia con la siguiente fórmula:
Test distance = (high level time x velocity of sound (340 M/S))/2



- Servomotor MG996R

Utiliza 3 pines: 1 para recibir una tensión de 5V (VCC), uno para tierra (GND) y otro para recibir señales de control. A través del pin de control recibe una señal PWM digital, que interpretará la frecuencia de los pulsos como distintos ángulos en sentido horario o antihorario, y girará para alcanzar ese ángulo.



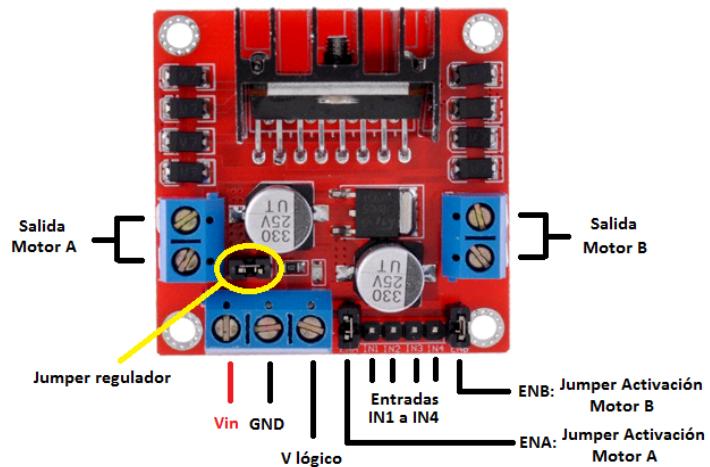
- Motores DC (Corriente Continua) RS-365 12V

Motores que girarán de acuerdo a la polaridad de sus bornes. Poseen dos bornes, (+) y (-). Si se ingresa una tensión en polaridad directa (tensión en borne positivo mayor que en borne negativo), el motor girará en sentido horario, y si se ingresa en polaridad inversa (tensión en borne negativo mayor que en borne positivo), girará en sentido antihorario. Haciendo uso de un puente H se puede regular la tensión recibida para hacer que gire en un sentido o en otro, y que su velocidad cambie cambiando el valor medio de la tensión.



- [Motor Shield L298N](#)

Circuito integrado que incluye un puente H de dos canales, para poder controlar los motores DC. Requiere ser alimentado con una tensión continua desde el pin Vin. En nuestro caso, se utilizaron baterías AA de 24 V. Para controlar la tensión enviada a los motores, se utiliza el V lógico, que recibe señales PWM.



- Luces LED

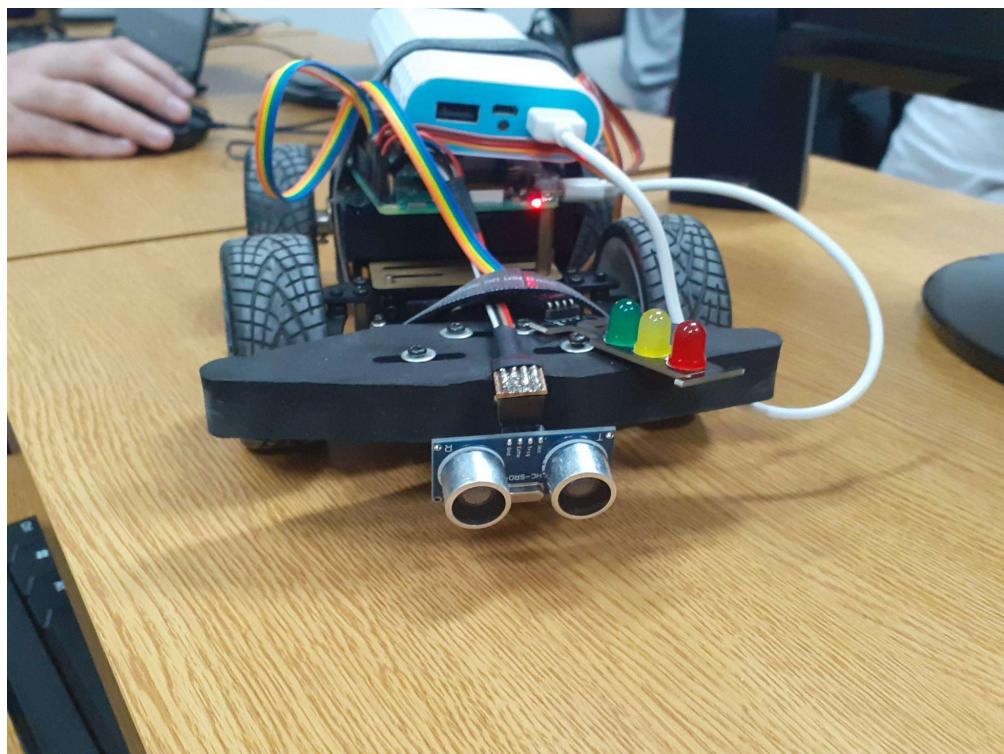
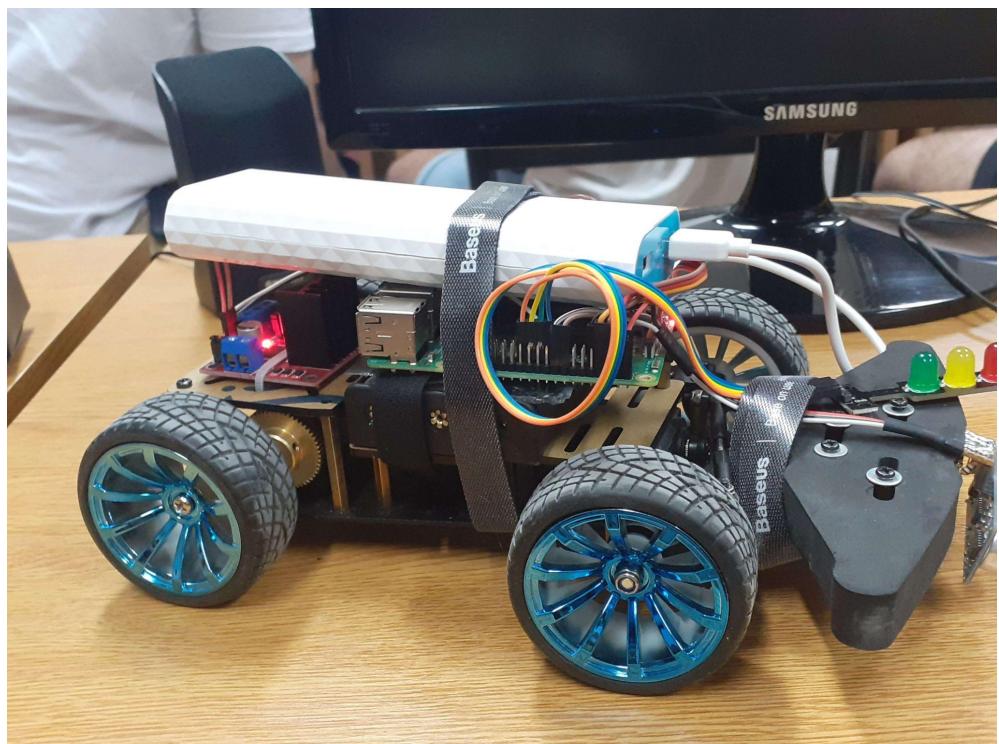
Tres LEDs, uno verde, uno amarillo y uno rojo, conectados en cátodo común, por lo que estos serán activos, se encenderán, cuando la señal de entrada de cada LED sea activa.



- Power Bank TP-Link 15600

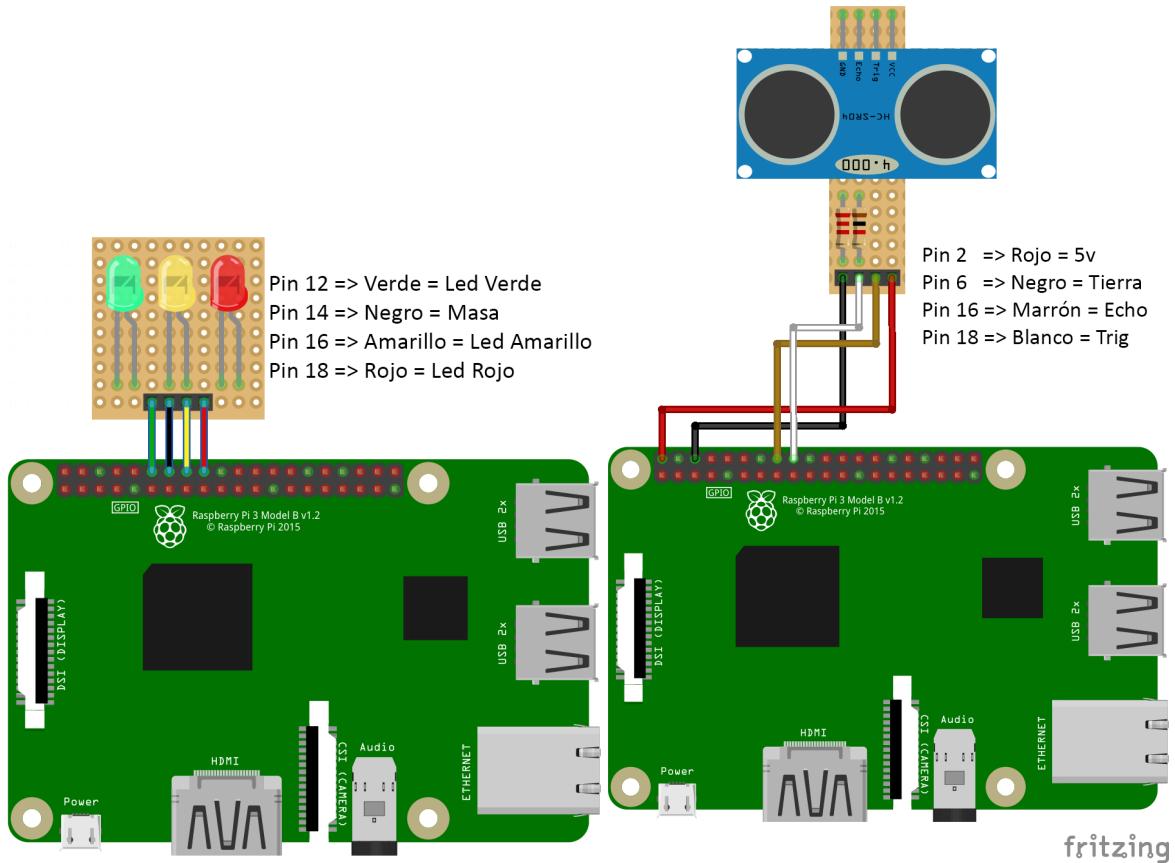


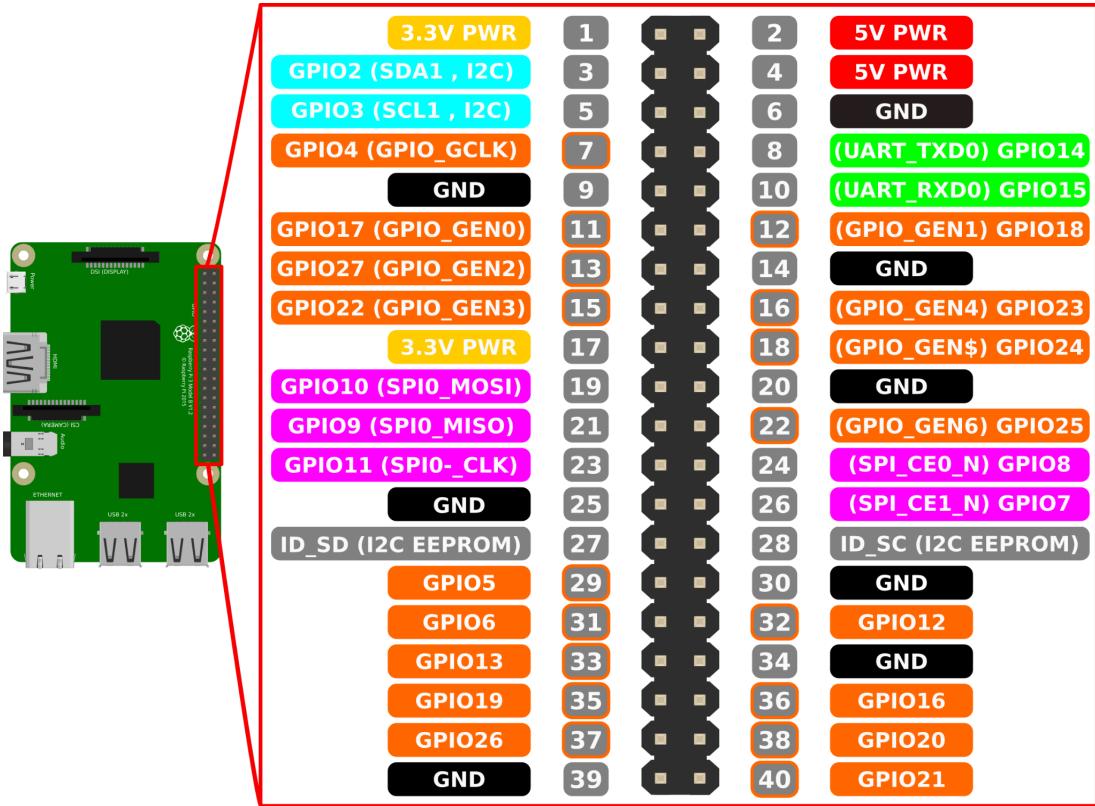
- Vehículo no tripulado con las conexiones realizadas



2.1.1 Conexiónado de elementos

Para realizar la conexión de los periféricos mostrados en el apartado anterior, utilizamos unas imágenes brindadas por la cátedra, donde se muestran los pines que debe utilizar cada uno y la manera en que deben estar posicionados en la Raspberry Pi. Además, empleamos una imagen que señalaba en detalle cada uno de los pines, su número y función. Sin embargo, para el sensor de ultrasonido, utilizamos los pines 40 para el trigger y 38 para el echo, ya que los sugeridos estaban siendo utilizados por las luces led.

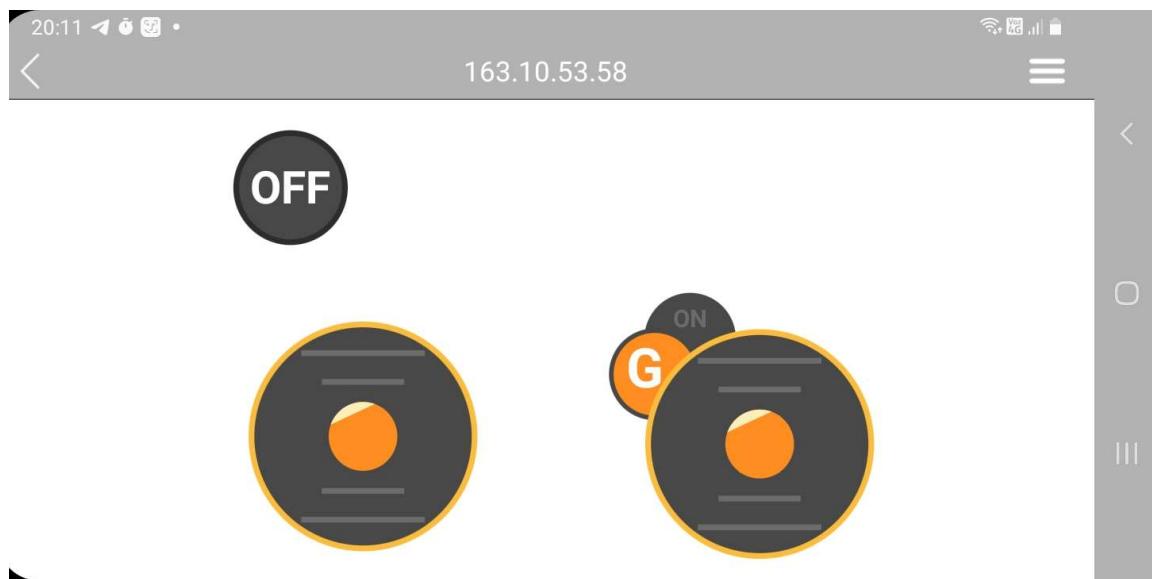




2.2 Software

- RemoteXY

RemoteXY es un programa que permite crear y utilizar una interfaz gráfica de usuario que fue utilizada para controlar el manejo del vehículo a través de un teléfono inteligente. La interfaz se realiza en la página y cuando están todos los elementos deseados, se obtiene un código que deberá ser insertado en uno de los nodos de node-red. Luego, se baja la aplicación móvil y se selecciona una conexión mediante “Ethernet” y se ingresa la ip de la raspberry. Al hacer este último paso, se verá una interfaz como la siguiente:



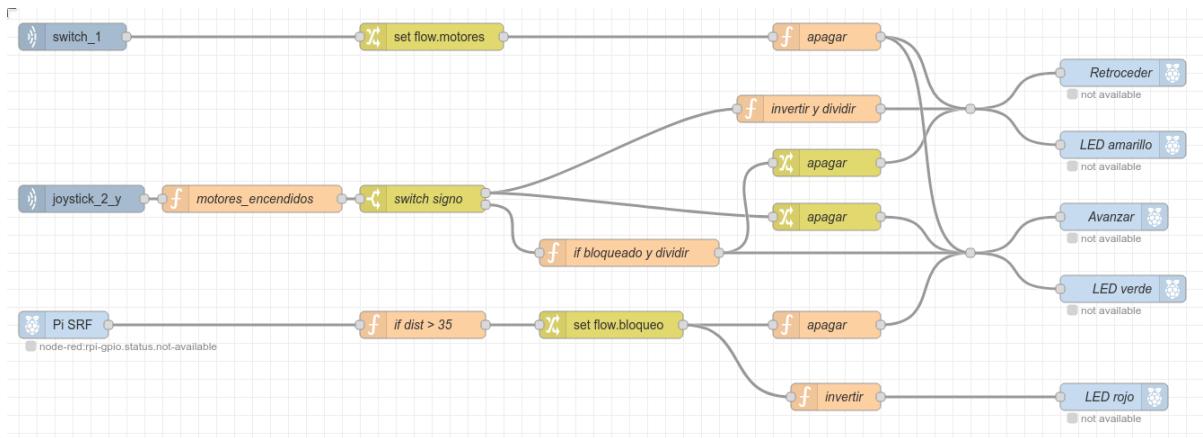
El botón de la parte superior simula el funcionamiento del arranque del motor, en caso de presionar “ON” se permite avanzar o retroceder y cambia su visualización a “OFF”, caso contrario el vehículo permanecerá en el lugar. Los joysticks permiten ir hacia adelante o hacia atrás, y girar a izquierda o derecha. Además, en el caso del joystick derecho, al mover la “G” hacia “ON”, se habilita la opción para controlar el vehículo con el acelerómetro del dispositivo móvil.

- Node-RED

Node-RED es una herramienta para desarrollo de soluciones IoT haciendo uso de diagramas de flujos, desarrollada por IBM junto a la JS Foundation. Permite crear, editar y desplegar aplicaciones distribuidas desde el navegador, las cuales se ejecutan sobre el runtime system Node.js. Los flujos consisten de nodos que representan la entrada o salida de mensajes en un dispositivo, o un procesamiento que se realiza sobre los mensajes, conectados entre sí e interactuando de acuerdo al protocolo de comunicaciones M2M (machine to machine) conocido como MQTT.

Se instalaron los siguientes módulos para:

- node-red-node-pi-gpio: leer y escribir en los pines de GPIO de la Raspberry Pi.
- node-red-node-pisrf: leer la distancia del sensor de proximidad HC-SR04.
- node-red-node-pi-gpiod: tener un nodo de salida de GPIO de la Raspberry Pi, en modo servo, con el cual el mensaje que toma tiene como información el ángulo en el que debe posicionarse.
- node-red-contrib-remote-xy: tener una interfaz en el dispositivo móvil.



En el primer flujo tenemos tres entradas:

- switch_1: un switch que se encuentra en la interfaz de Remote XY.
- joystick_2_y: eje "y" de un joystick que se encuentra en la interfaz de Remote XY.
- Pi SRF: el sensor de ultrasonido HC-SR04 conectado a la Raspberry Pi.

También hay cinco salidas:

- Retroceder: una de las entradas de control del puente H L298, que habilita la polarización de los motores para retroceder. Configurado como salida PWM.
- Avanzar: una de las entradas de control del puente H L298, que habilita la polarización de los motores para avanzar. Configurado como salida PWM.
- LED amarillo: cátodo de un LED amarillo conectado a la Raspberry Pi. Configurado como salida PWM.
- LED verde: cátodo de un LED verde conectado a la Raspberry Pi. Configurado como salida PWM.
- LED rojo: cátodo de un LED rojo conectado a la Raspberry Pi. Configurado como salida digital.

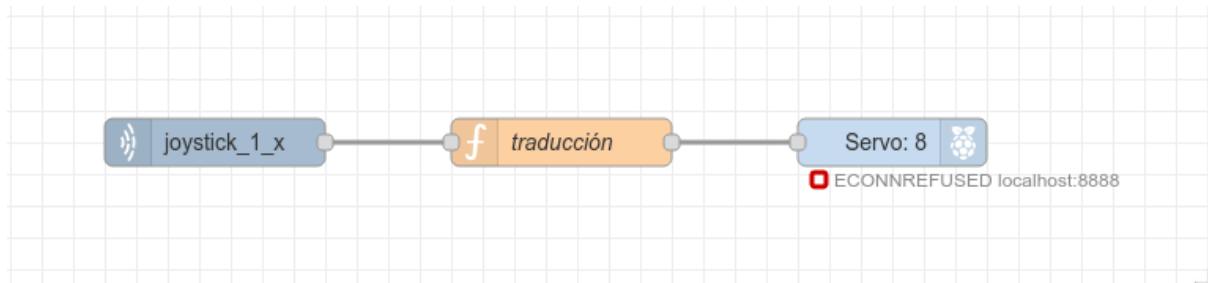
La entrada "switch_1" ingresa mensajes del tipo booleano. Dicho mensaje setea un contexto de flujo "motores", que habilita o deshabilita dichos componentes. Luego, si el mensaje es "false", envía un 0 a las salidas "Retroceder", "LED amarillo", "Avanzar" y "LED verde".

La entrada "joystick_2_y" ingresa mensajes del tipo enteros en un rango de -100 a 100. Dicho mensaje se descarta si el contexto de flujo "motores" está en falso. Si no se descarta, pasa por un nodo que divide en dos salidas dependiendo si el mensaje es negativo o positivo:

- Si es negativo, se divide en dos ramas. En la primera invierte el signo del mensaje y lo decrementa a la mitad para luego enviarlo a las salidas "Retroceder" y "LED amarillo". El decremento sucede ya que si los motores operan al 100% resultan muy difíciles de controlar. En la segunda rama, envían un 0 a las salidas "Avanzar" y "LED verde".
- Si es positivo, primero se descarta el mensaje si el contexto de flujo "bloqueo" es falso. Si no lo es, se decrementa a la mitad y se divide en dos ramas. En la primera se envía un 0 a las salidas "Retroceder" y "LED amarillo". En la segunda, envía el mensaje a las salidas "Avanzar" y "LED verde".

La entrada "Pi SRF" ingresa mensajes del tipo enteros, donde el valor indica la distancia medida por el sensor de ultrasonido en centímetros. Dichos mensajes los recibe un nodo que si la distancia es mayor a 35 centímetros, envía un verdadero; si no, envía un falso.

Luego setea el contexto de flujo “bloqueo” como el mensaje que recibe. A continuación se divide en dos ramas. En la primera, envía un 0 a las salidas “Avanzar” y “LED verde”. En la segunda envía el mensaje invertido a la salida “LED rojo”.



El segundo flujo tiene como entrada “joystick_1_x”, que se trata del eje “x” de un joystick definido en la interfaz de Remote XY. Este ingresa mensajes de tipo enteros de -100 a 100. Estos deben traducirse al rango del nodo del servo (0 - 100). Entonces, al mensaje se le suma 100 y luego se lo divide por 2.

3 - Demostración del funcionamiento

Se adjunta a continuación un video del funcionamiento del vehículo.

<https://drive.google.com/file/d/1CG3XiFr-QN2RSQzyOLIbHtT-tvuSLg/view?usp=sharing>

Fuentes

- Imágenes de las pruebas realizadas:
https://drive.google.com/drive/u/0/folders/1_wsbBQPiyz6N0PaKXBvXf8xh8EZWIe8
- Datasheet del Raspberry PI 3 Model B:
<https://static.raspberrypi.org/files/product-briefs/Raspberry-Pi-Model-Bplus-Product-Brief.pdf>
- Datasheet del sensor HC-SR04:
<https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf>
- Datasheet del servomotor MG996R:
https://www.electronicoscaldas.com/datasheet/MG996R_Tower-Pro.pdf
- Datasheet L928N:
https://www.sparkfun.com/datasheets/Robotics/L298_H_Bridge.pdf
- Datasheet motor DC RS-365:
<https://5ororwxhiiqojij.leadongcdn.com/LS-RS-365-aidliBqmKinSRqqnijrlnj.pdf>
- Página oficial de Node-RED: <https://nodered.org/>
- Página oficial de RemoteXY: <https://remotexy.com/>