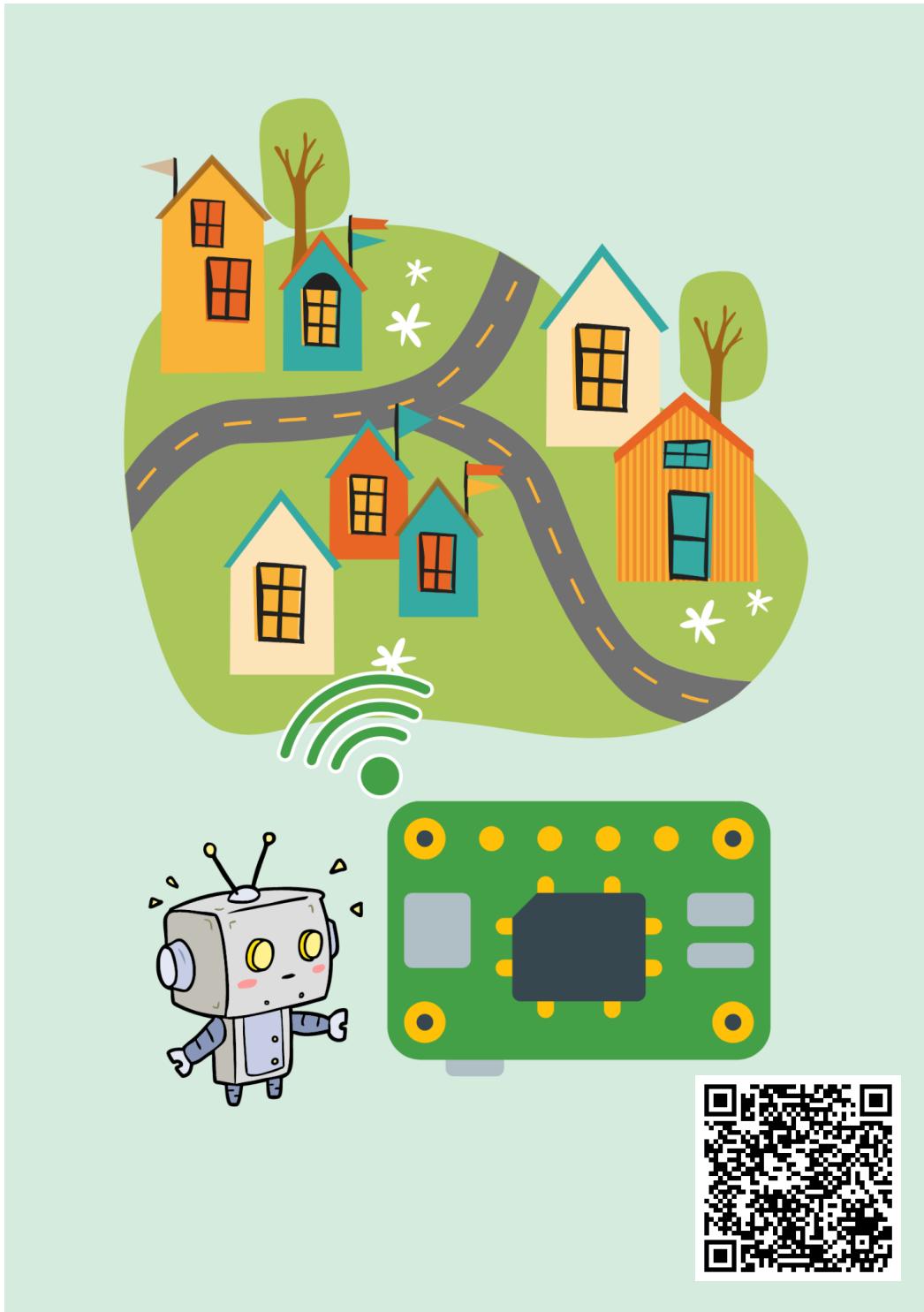


Greenbots: La Ciudad Inteligente



Autores: María Godoy, Valeria Meléndez, Raúl Baladez, Abderrahman Ben Hammout, Amira Khadra, Javier Barragán, Sergio Liñán, Miguel Ángel Blanch, Sara Gallego, Alejandro Señor, Iván Gutiérrez, Sofía Taja, José Martino, Héctor Torres, Juan Carlos Alcalde y Seraj El Karrichi coordinados por José López

Índice

Índice	2
1. Resumen	3
2. Abstract	3
3. Introducción	4
4. Desarrollo	6
4.1. Tecnología Empleada	6
4.2. Proyectos que componen el trabajo	6
4.2.1. Semáforo de control del tráfico	6
4.2.2. Semáforo control de contaminación	14
4.2.3. Panel inteligente	14
4.2.4. Coche Inteligente	16
4.2.5. Huerto Inteligente	18
4.3. Esquema final del proyecto	19
5. Conclusión	20
6. Bigliografía/Webgrafía	20

1. Resumen

Este proyecto ha sido realizado por alumnado del IES El Getares de Algeciras (Cádiz), principalmente de la asignatura de Computación y Robótica de los cursos 2º y 3º de ESO. Hemos empleado la robótica y el IoT combinando diferentes tecnologías para crear prototipos de sistemas que nos ayuden a mejorar nuestra salud, la de nuestras ciudades y la del planeta.

El proyecto se denomina Greenbots. El nombre es el resultado de combinar los términos Green y Bots, haciendo alusión al objetivo principal del proyecto: conseguir que la robótica nos ayude a tener ciudades más verdes.

Las tecnologías empleadas han sido: Raspberry Pi, Arduino, Micro:bit, Scratch, Arduino IDE, TinkerCAD y Microsoft MakeCode. El trabajo ha sido el resultado de combinar diferentes proyectos desarrollados en equipos, que de manera conjunta han permitido construir Greenbots, la ciudad del futuro.

2. Abstract

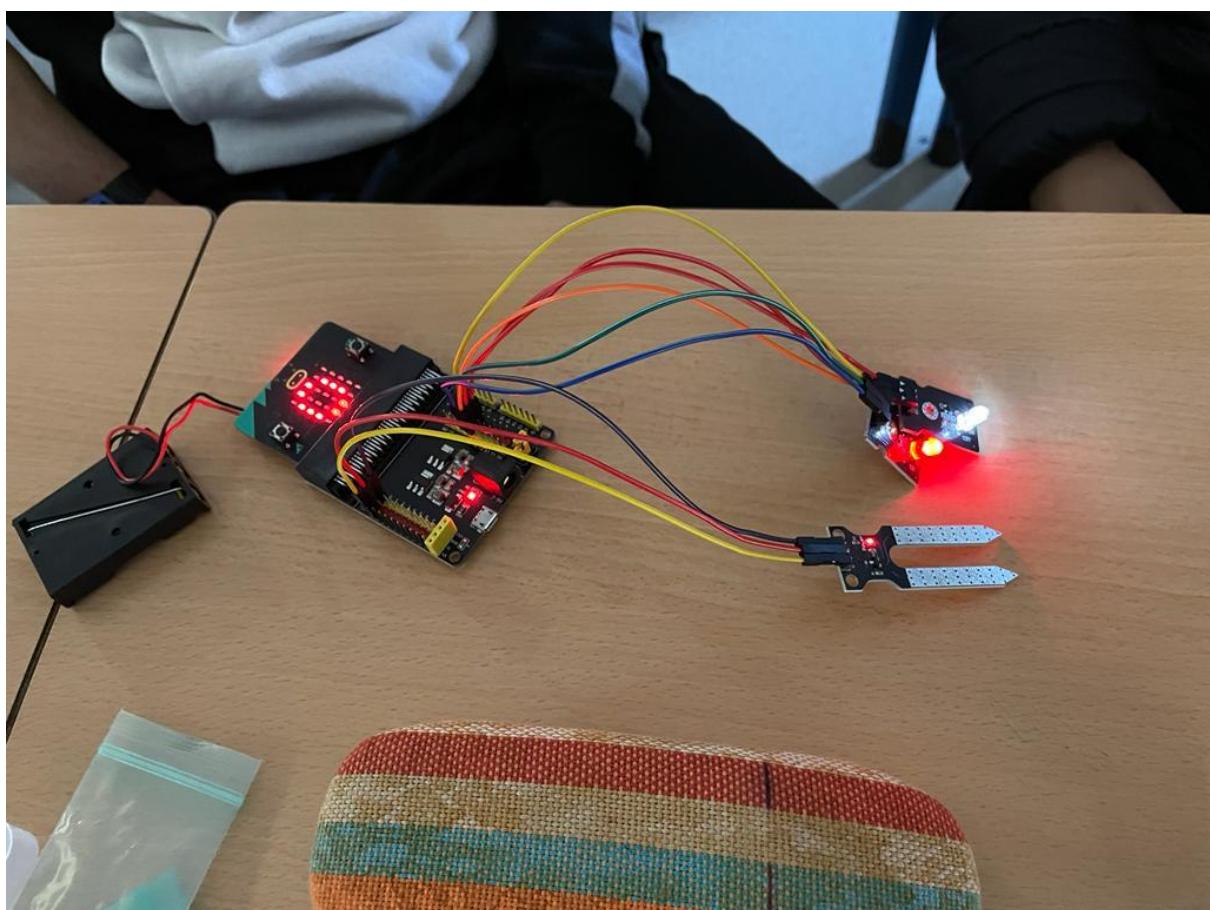
This project has been developed by students of IES El Getares located in Algeciras (Cádiz), mainly from the Computing and Robotics subject taught both in 2ESO and 3ESO levels. We have used robotics and IoT combining different technologies to create prototype systems that help us improve our health, and the health of our cities and of the planet.

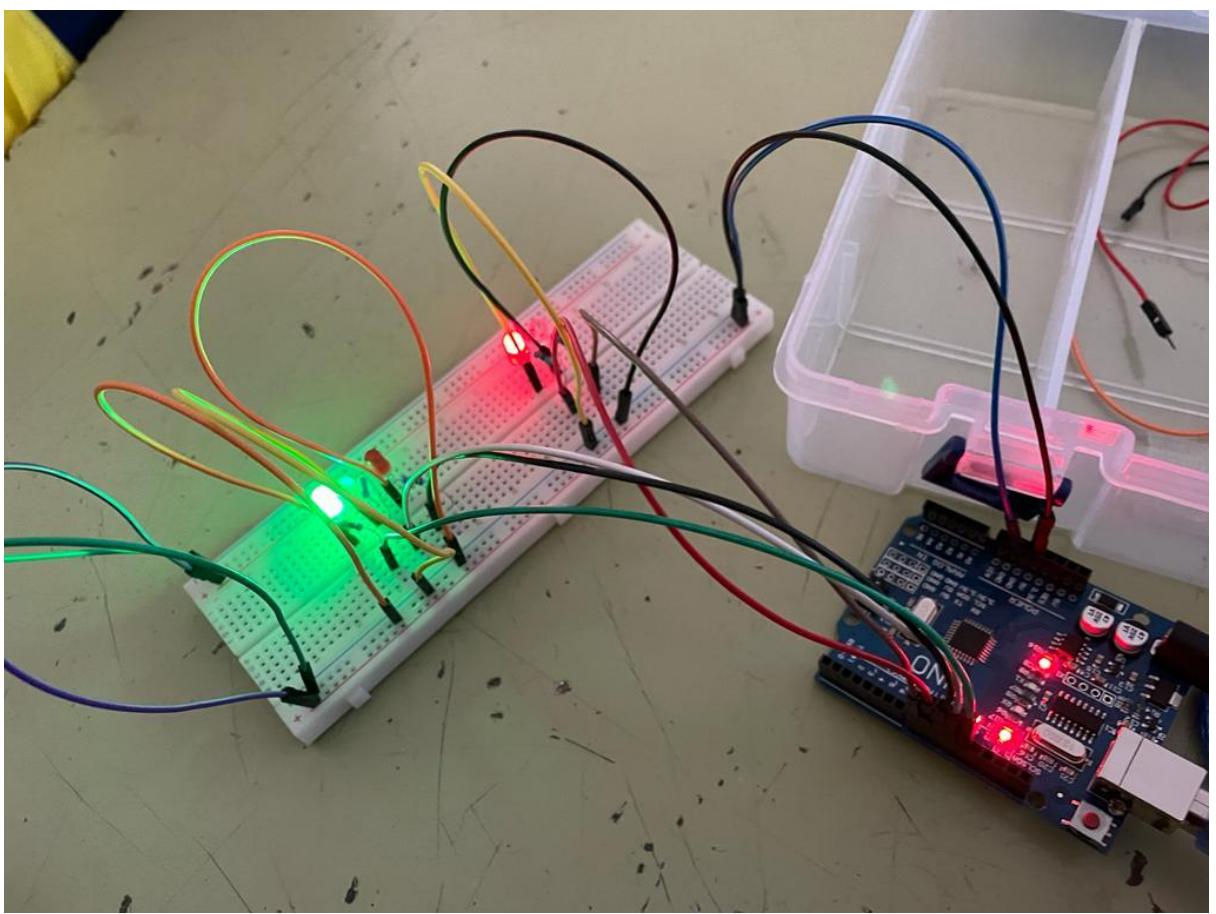
The project is called Greenbots. The name is the result of combining the terms Green and Bots, alluding to the main objective of the project: using robotics to help us having greener cities.

The technologies used have been: Raspberry Pi, Arduino, Micro:bit, Scratch, Arduino IDE, TinkerCAD and Microsoft MakeCode. The work has been the result of combining different projects developed in teams, which together have made it possible to build Greenbots, the city of the future.

3. Introducción

Este trabajo se ha desarrollado principalmente durante el segundo trimestre del presente curso 21/22. En la asignatura Computación y Robótica de 2º y 3º de ESO encontramos los bloques de contenidos de IoT y Robótica respectivamente. Para trabajar estos bloques se han desarrollado en equipo proyectos de forma colaborativa. Al finalizar el segundo trimestre se han seleccionado los mejores proyectos de ambos cursos para poder combinarlos y realizar un macroproyecto que permita construir una ciudad inteligente en equipo: Greenbots.





4. Desarrollo

4.1. Tecnología Empleada

- Hardware
 - Raspberry Pi
 - Arduino
 - Micro:bit
- Software:
 - Microsoft MakeCode
 - Scratch
 - Tinkercad
 - Arduino IDE

4.2. Proyectos que componen el trabajo

- Semáforos inteligentes:
 - Control contaminación: sensores de análisis del aire que cortan el tráfico cuando la contaminación es alta
 - Control del tráfico: control de tráfico para restringirlo cuando hay demasiado
- Panel inteligente: Mostrando información en tiempo real a los conductores
- Coche inteligente: Conducción autónoma, y sistema start-stop que detecta si el tráfico está abierto o no.
- Huerto inteligente: Sistema de regado e iluminación automática en el huerto.

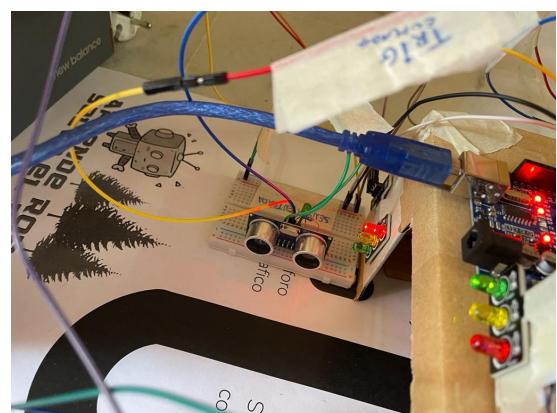
Durante el inicio del tercer trimestre, el alumnado de 2º bachillerato ha realizado la auditoría de los proyectos, y una descripción de su funcionamiento, componentes y resultados tal y como viene recogido en los siguientes puntos:

4.2.1. Semáforo de control del tráfico

Descripción:

El objetivo del proyecto es crear un dispositivo capaz de controlar el número de coches en la ciudad. Para ello podremos cerrar y abrir el tráfico de vehículos mediante un semáforo el cual está conectado a dos sensores de ultrasonidos, capaces de detectar el número de coches que entran y salen de la ciudad.

El número máximo de coches que pueden estar a la vez en la ciudad son 11. Cuando los coches

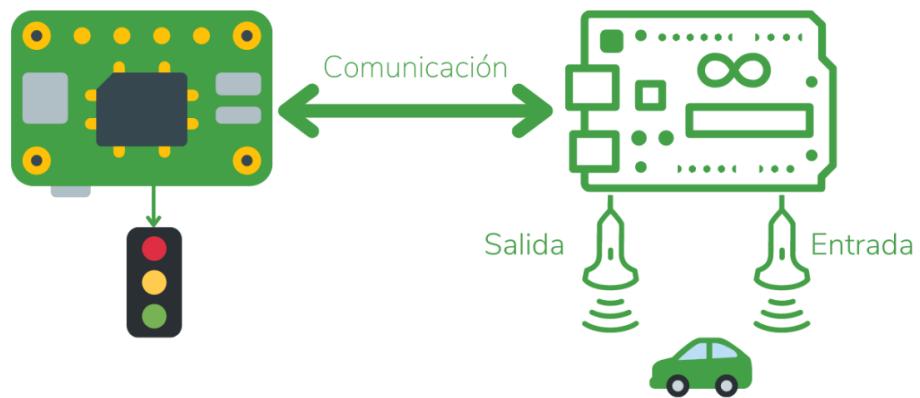


igualan este número el semáforo cambia al color rojo y volverá al verde cuando el detector de ultrasonidos encuentre a un coche saliendo.

Componentes:

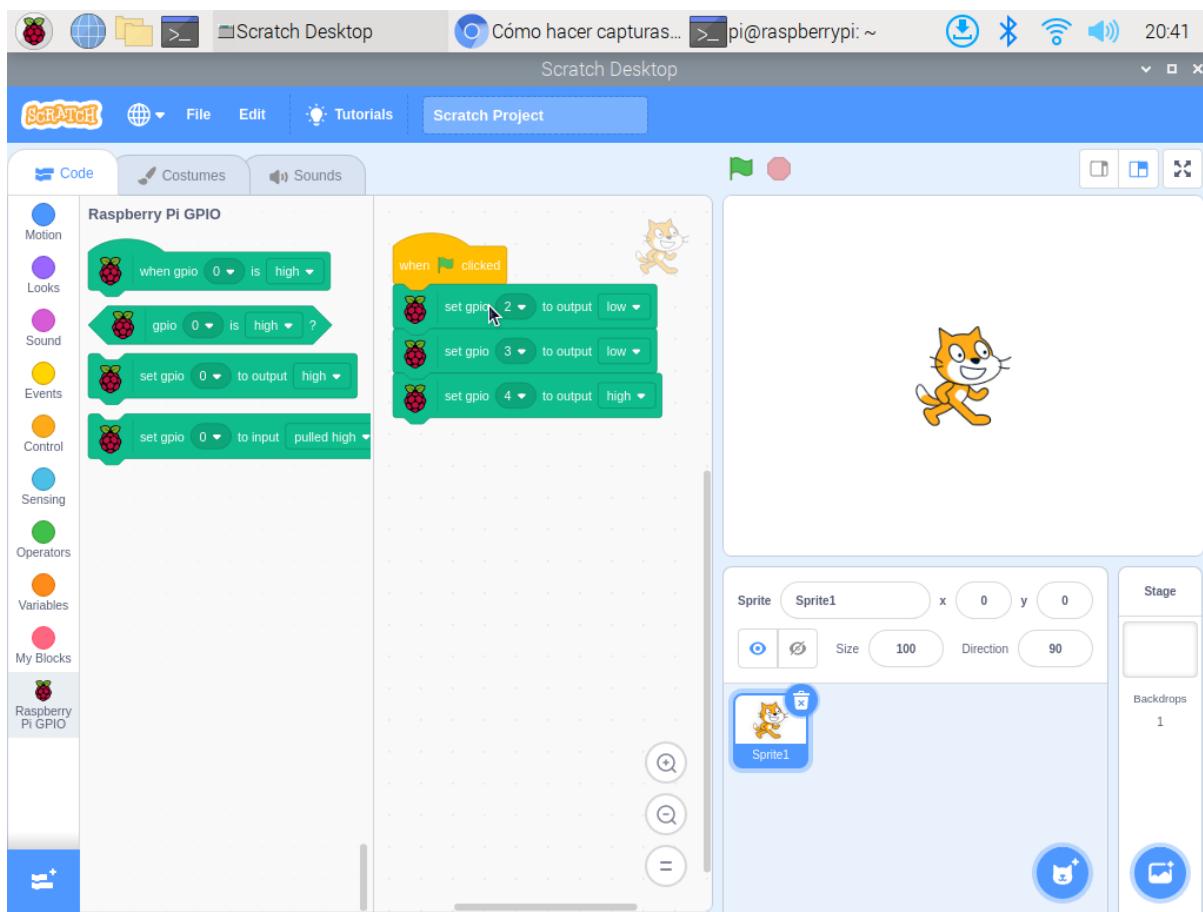
- Raspberry Pi
- 2x Sensores de ultrasonido
- Semáforo
- Arduino

Esquema de funcionamiento:

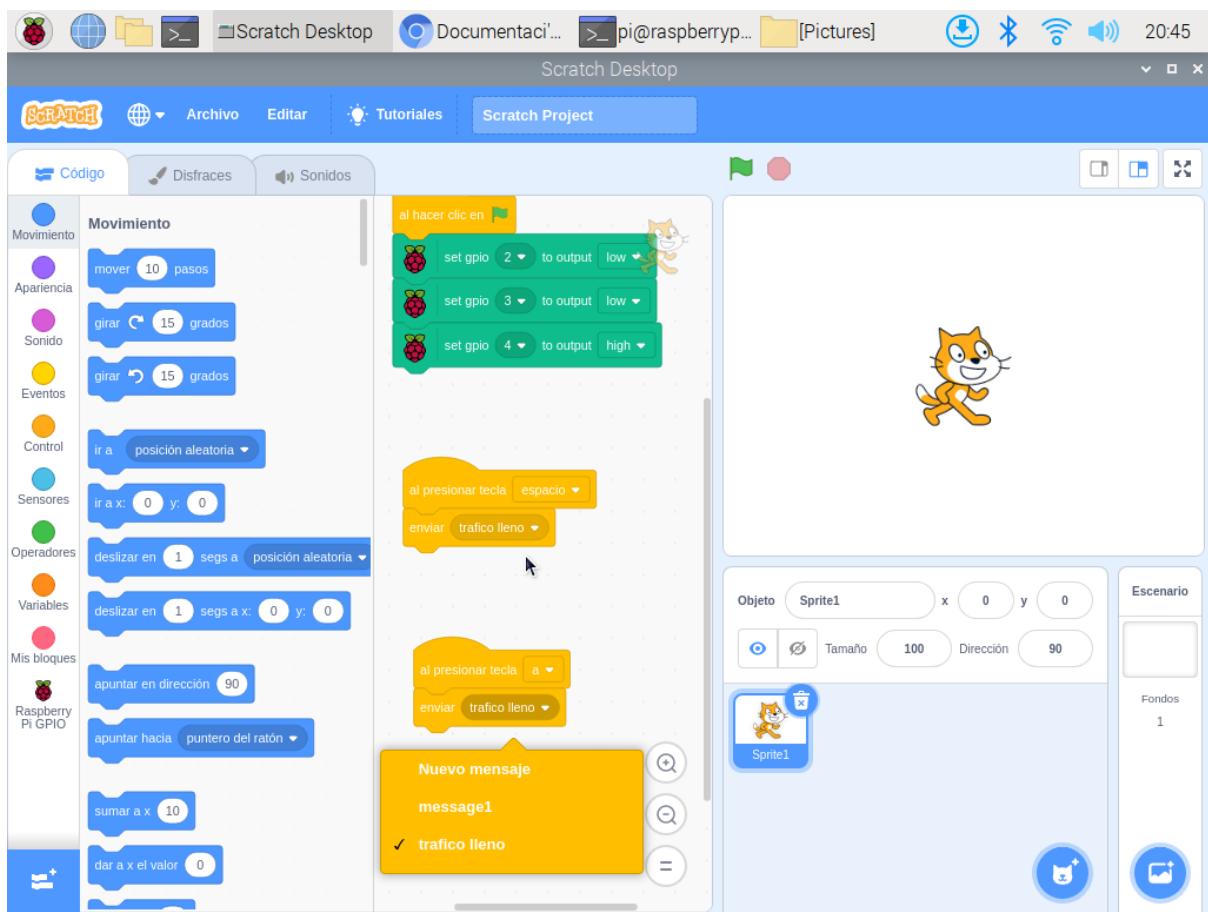


Software:

- **Control del semáforo (Scratch):**

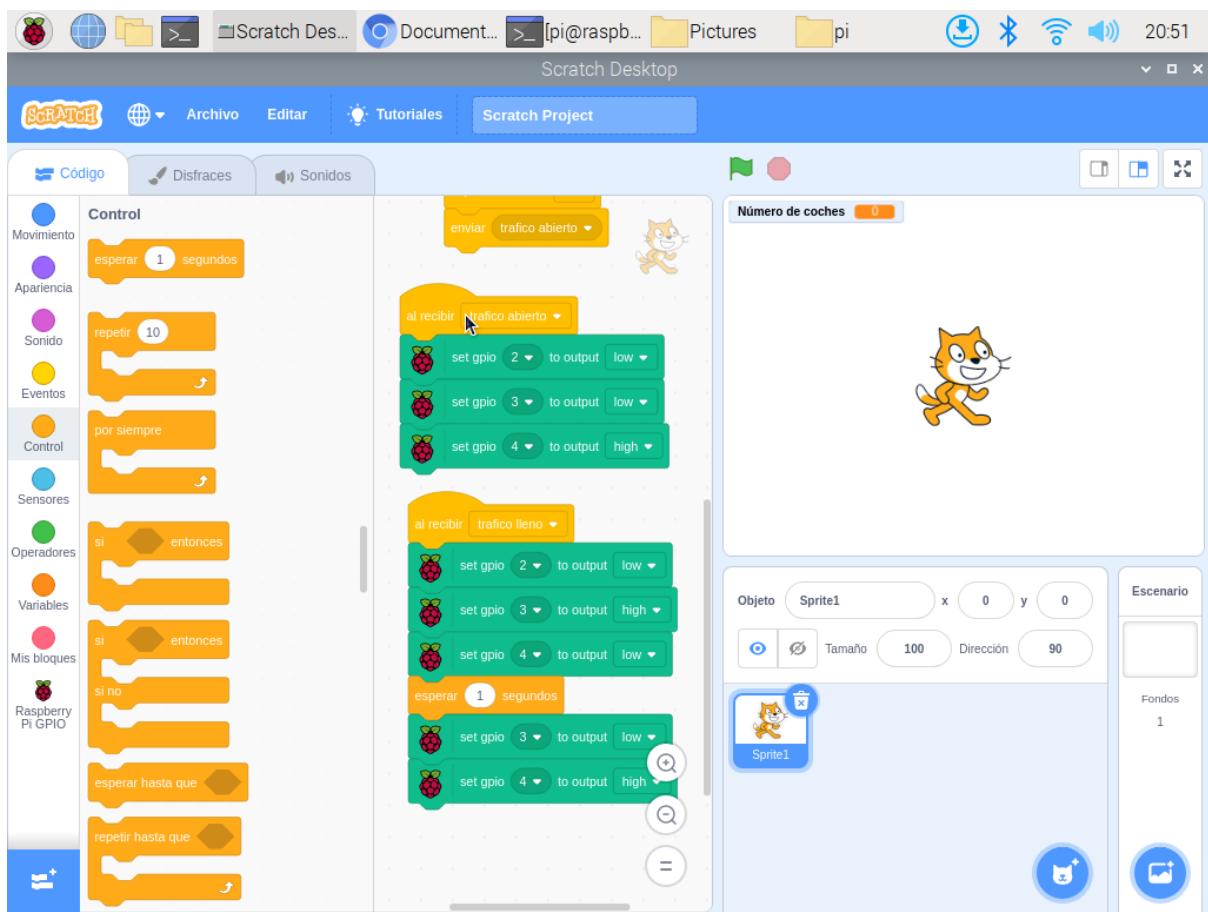


Creamos eventos (los eventos nos van a ayudar a controlar todas las acciones de nuestro robot.



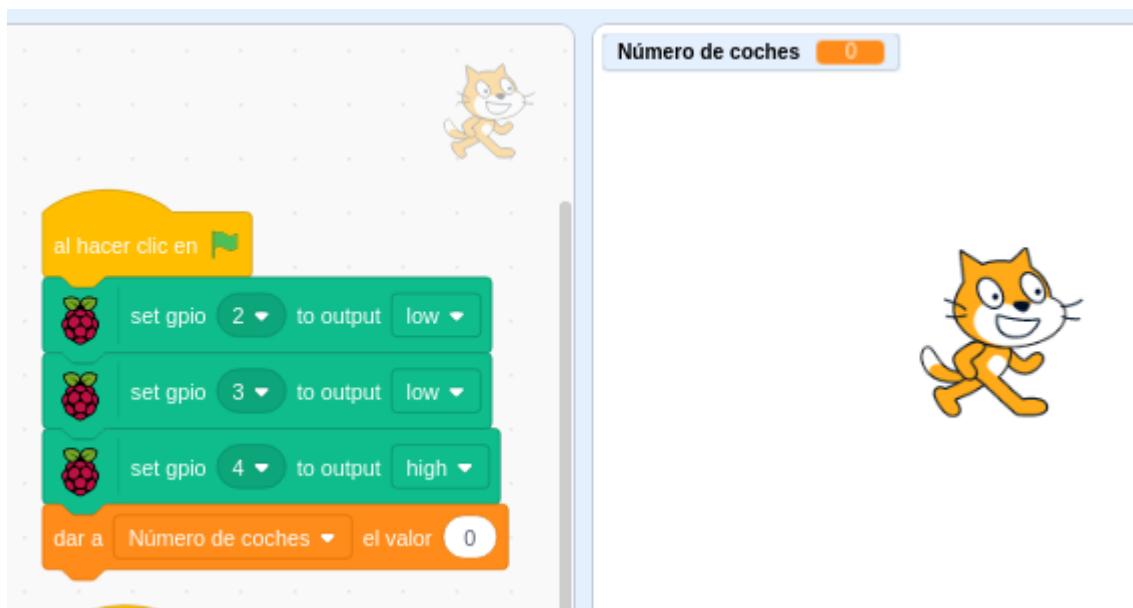
Un evento que lanzamos cuando el tráfico está lleno, y otro para cuando está libre. En cada caso pondremos el semáforo en rojo o en verde.

Capturamos esos eventos y cuando se producen, cambiamos el semáforo al estado que corresponda:



De esta forma, se pone en rojo cuando el tráfico está lleno y verde cuando está libre. Aunque lo estamos simulando. En nuestro proyecto los eventos se deben lanzar cuando contemos los coches que pasen.

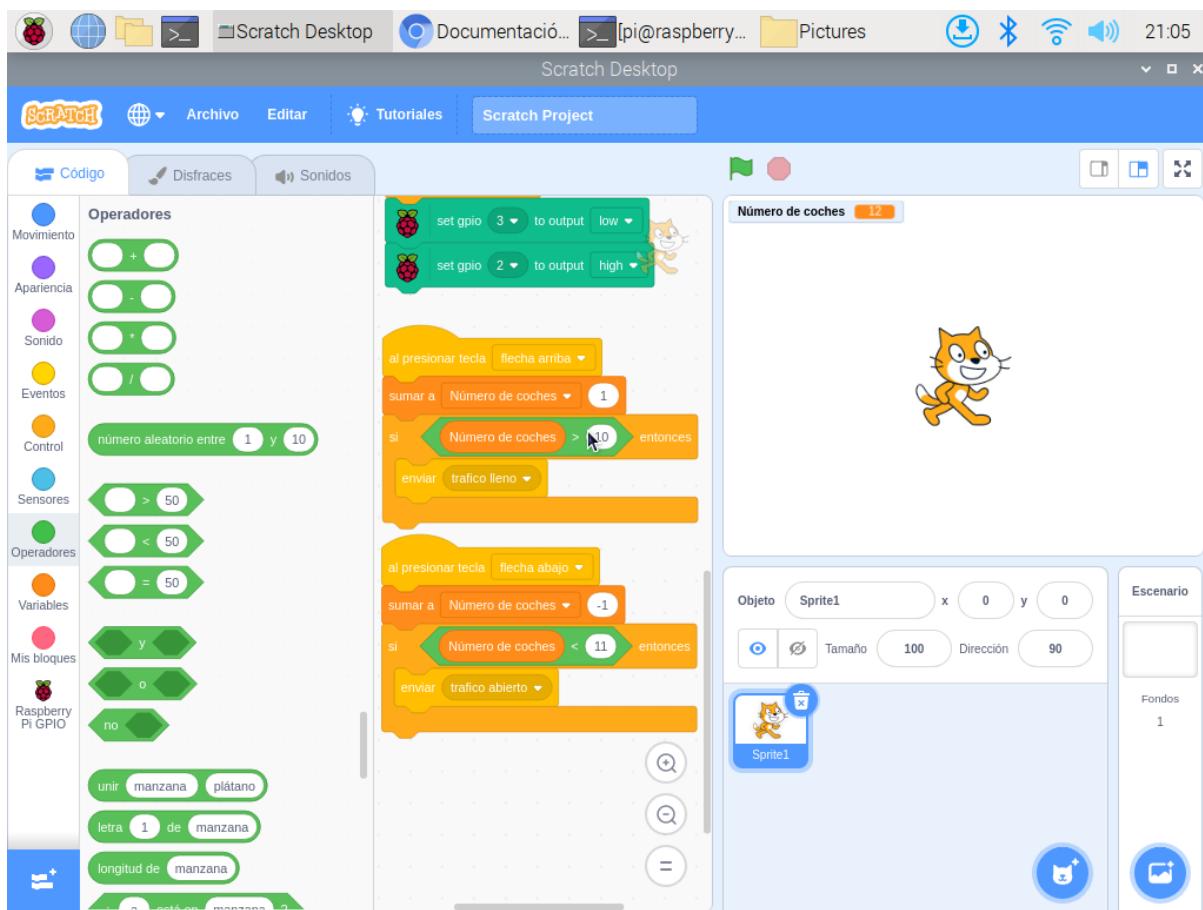
Vamos a crear una variable que se llame “Número de coches” que cuente los coches que hay en cada momento. La inicializamos con 0 al pulsar la bandera:



y nuevamente, vamos a simular el paso de coches con el teclado. Tecla arriba submimos coches, y tecla bajo, restamos coches:



Y vamos a controlar de forma que si hay más de 10 coches, se lance el mensaje de Tráfico lleno, y si no el de tráfico libre. Además ya podemos borrar los bloques de código donde controlábamos cuando se lanzaban los mensajes de tráfico abierto-cerrado con el teclado:



Ahora tan sólo debemos cambiar para que los eventos de "Tráfico lleno" y "Tráfico libre" se envíen cuando se reciba una señal digital procedente del arduino que controla los sensores de ultrasonidos que cuentan el tráfico.

- Contador de tráfico (Arduino):

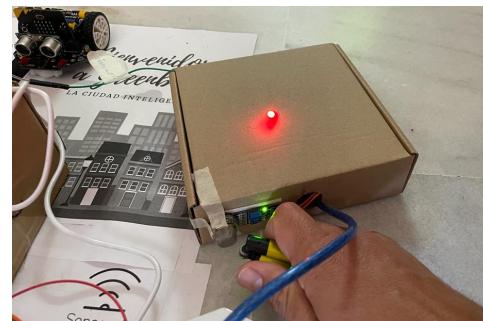
Debemos contar los coches que salen para sumar un nuevo coche, y los que salen para restar otro. Para ello tendremos en cuenta los objetos que pasan a menos de 10 cms del sensor de ultrasonidos y usaremos la misma función de controlador de sensor. Cuando se supeere el número de 11 coches, se enviará una señal digital a la Raspberry para indicar que el tráfico está lleno, y en caso contrario otra señal para indicar que el tráfico está libre. El código es el siguiente:

```
void setup() {
    Serial.begin(9600);
    pinMode(3, OUTPUT);
    pinMode(6, OUTPUT);
    pinMode(4, OUTPUT);
    pinMode(5, OUTPUT);
    digitalWrite(5, HIGH);
    digitalWrite(4, LOW);
}
void loop() {
    distancia = 0.01723 * readUltrasonicDistance(3, 2);
    Serial.println(distancia);
    if (distancia < 10) {
        digitalWrite(3, LOW);
        N_coches += 1;
        delay(2500); // Wait for 2500 millisecond(s)
        digitalWrite(3, HIGH);
    }
    distancia_salida = 0.01723 * readUltrasonicDistance(6, 7);
    Serial.println(distancia_salida);
    if (distancia_salida < 10) {
        digitalWrite(6, LOW);
        N_coches += -1;
        delay(2500); // Wait for 2500 millisecond(s)
        digitalWrite(6, HIGH);
    }
    if (N_coches >= 10) {
        digitalWrite(4, HIGH);
        digitalWrite(5, LOW);
    }
    if (N_coches <= 11) {
        digitalWrite(5, HIGH);
        digitalWrite(4, LOW);
    }
    Serial.print("Número de coches: ");
    Serial.println(N_coches);
}
```

4.2.2. Semáforo control de contaminación

Descripción:

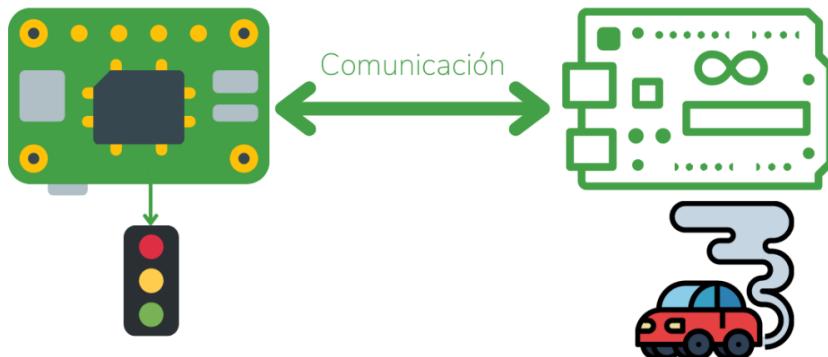
Esta parte del proyecto consta de un detector de gases construido mediante tecnología Arduino. Cuando detecta un determinado nivel de gases en la atmósfera (simulando una contaminación alta) manda una señal a la Raspberry Pi que ejecuta un programa de Scratch y ,manda una señal al semáforo para que se ponga en rojo y así avisarnos de este nivel de gas. Además el Arduino está conectado a un LED que se ilumina cuando detecta el nivel alto de gases.



Componentes:

- Raspberry Pi
- Sensor de gas MQ-2
- Semáforo
- Arduino

Esquema de funcionamiento:



Software:

Utilizamos un esquema de software muy similar al del proyecto anterior. Arduino enviará una señal digital a Raspberry cuando detecte un nivel de gas elevado, y Raspberry cuando reciba esa señal, activará las luces del semáforo que correspondan.

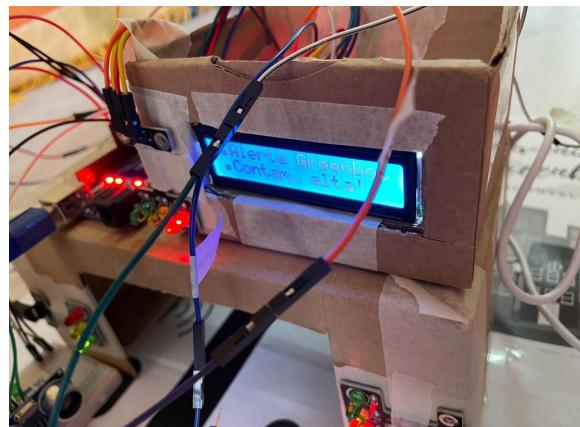
4.2.3. Panel inteligente

Descripción:

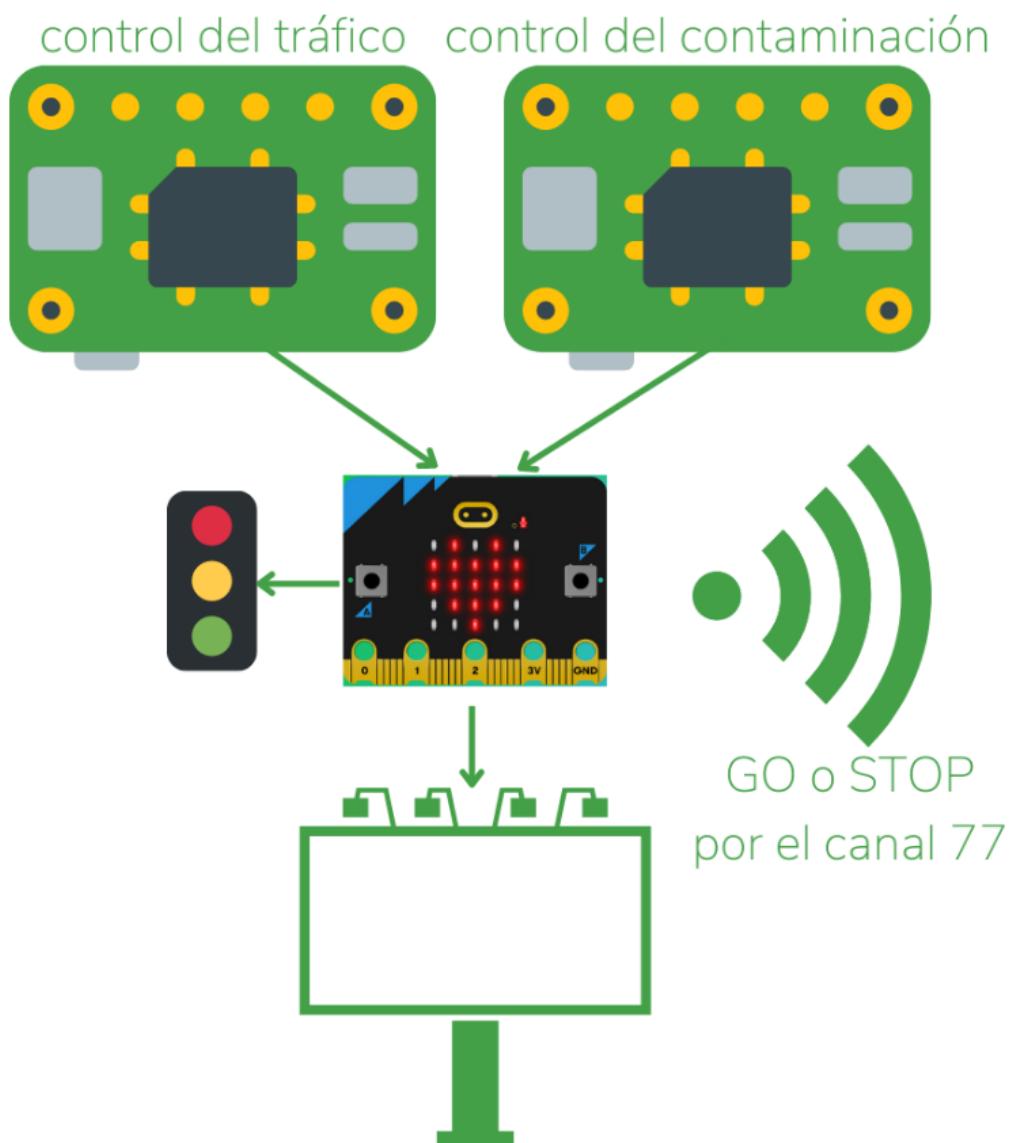
Este proyecto se encarga de recibir las señales de los proyectos de sensor de tráfico y sensor de contaminación, para informar a los conductores de la situación en tiempo real de la ciudad. Si la ciudad está libre de contaminación y sin saturación de tráfico, muestra un mensaje de que la ciudad está abierta y libre de humos, y enciende el semáforo verde. En caso contrario muestra las advertencias

correspondientes y activa la luz roja del semáforo. Está desarrollado con Micro:bit, y reciba entradas digitales a través de sus pines de los otros proyectos.

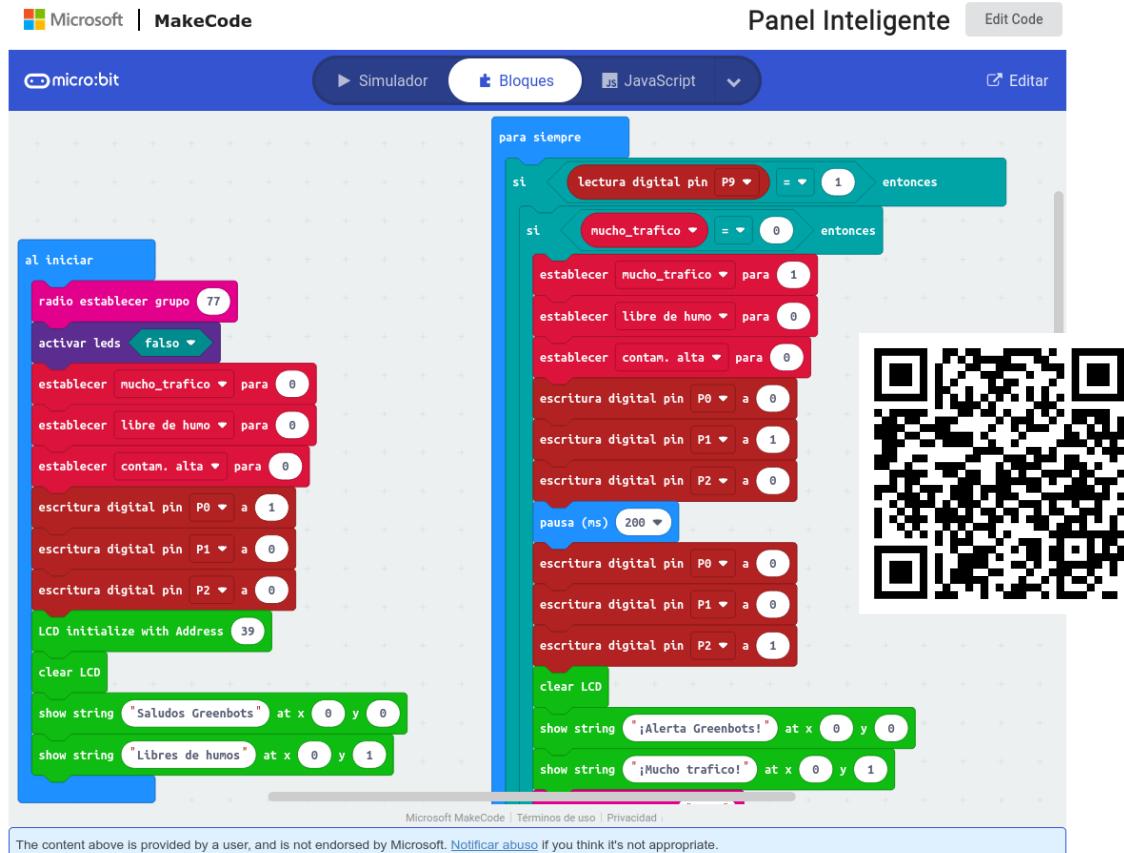
Además, el panel envía por radio a través del canal 77 señales inalámbricas con el mensaje GO o STOP dependiendo de si el tráfico está abierto o cerrado, de esta manera los coches pueden saber si pueden pasar o no a la ciudad.



Esquema de funcionamiento:



Software:

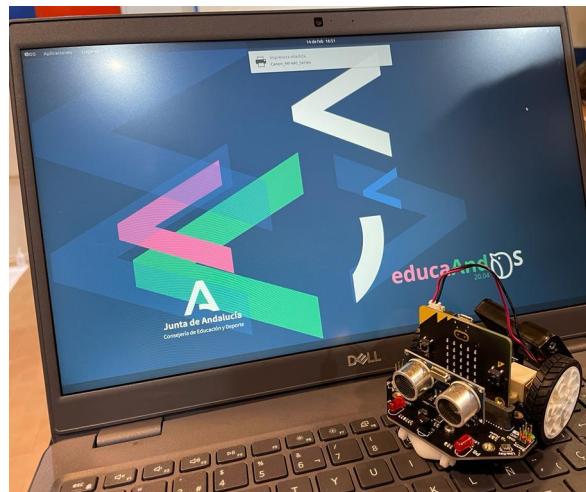


[Enlace al código](#)

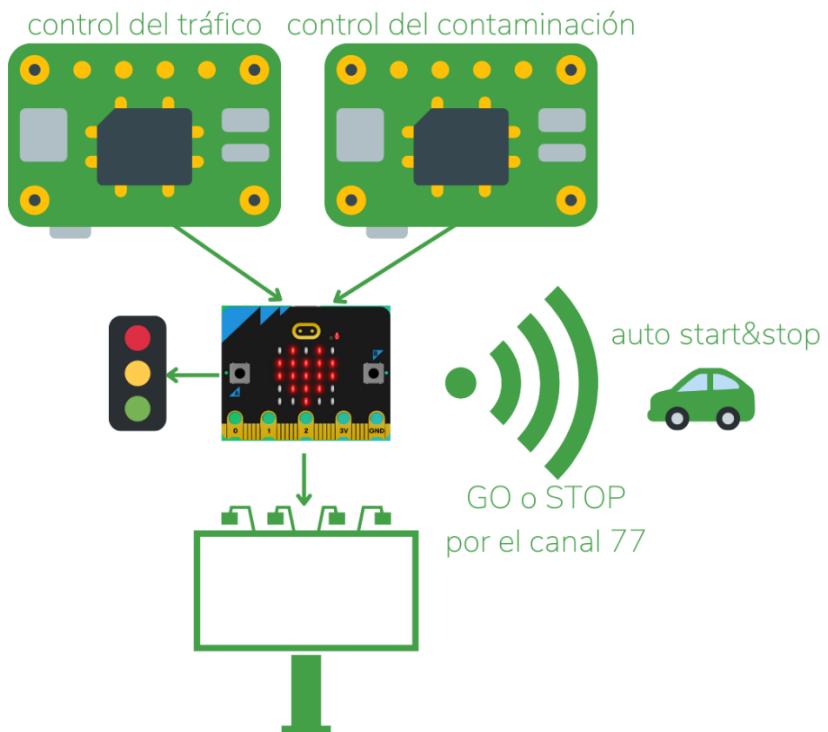
4.2.4. Coche Inteligente

Descripción:

Es un coche de conducción autónoma, creado con la tecnología de Micro:bit Maqueen. Este coche es capaz de seguir una línea negra con un fondo blanco (El clásico siguelíneas). La novedad es que el coche es capaz de detectar las señales de radio STOP y GO para avanzar o detenerse cuando el tráfico está cerrado o abierto:



Esquema de funcionamiento:



Software:

Microsoft | MakeCode

Coche McQueen

Edit Code

micro:bit

Simulador Bloques JavaScript Editar

para siempre

si Leer sigui linea izquierdo = 0 y Leer sigui linea derecho = 0 entonces

si estado = "GO" entonces

Motor izquierdo sentido avanzar velocidad 50

Motor derecho sentido avanzar velocidad 50

si no

Parar motor ambos

si Leer sigui linea izquierdo = 0 y Leer sigui linea derecho = 1 entonces

si estado = "GO" entonces

Motor izquierdo sentido avanzar velocidad 0

Motor derecho sentido avanzar velocidad 50

si no

Parar motor ambos

Microsoft MakeCode | Términos de uso | Privacidad | Notificar abuso

The content above is provided by a user, and is not endorsed by Microsoft. [Notificar abuso](#) if you think it's not appropriate.

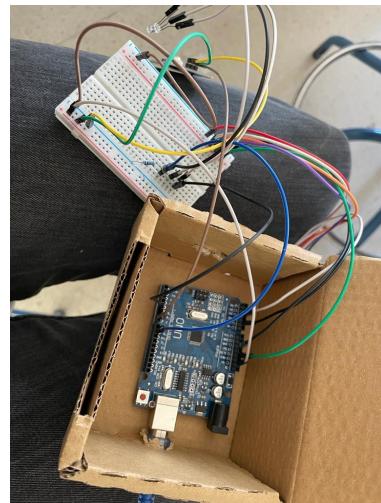
Enlace al código

A screenshot of the Microsoft MakeCode editor interface for a micro:bit project titled "Coche McQueen". The code is written in the "Bloques" (Blocks) language. It starts with a "para siempre" (forever) loop. Inside, there are two "si" (if) blocks. The first "si" block checks if both left and right line sensors are 0. If true, it checks if the state is "GO". If true, it moves both motors forward at 50% speed. If false, it stops both motors. The second "si" block checks if the left sensor is 0 and the right sensor is 1. If true, it checks if the state is "GO". If true, it moves the left motor forward at 0% speed and the right motor forward at 50% speed. If false, it stops both motors. At the bottom of the code, there are links to Microsoft MakeCode terms of use and privacy policy, and a "Notificar abuso" (Report abuse) link. A QR code is also present on the right side of the screen.

4.2.5. Huerto Inteligente

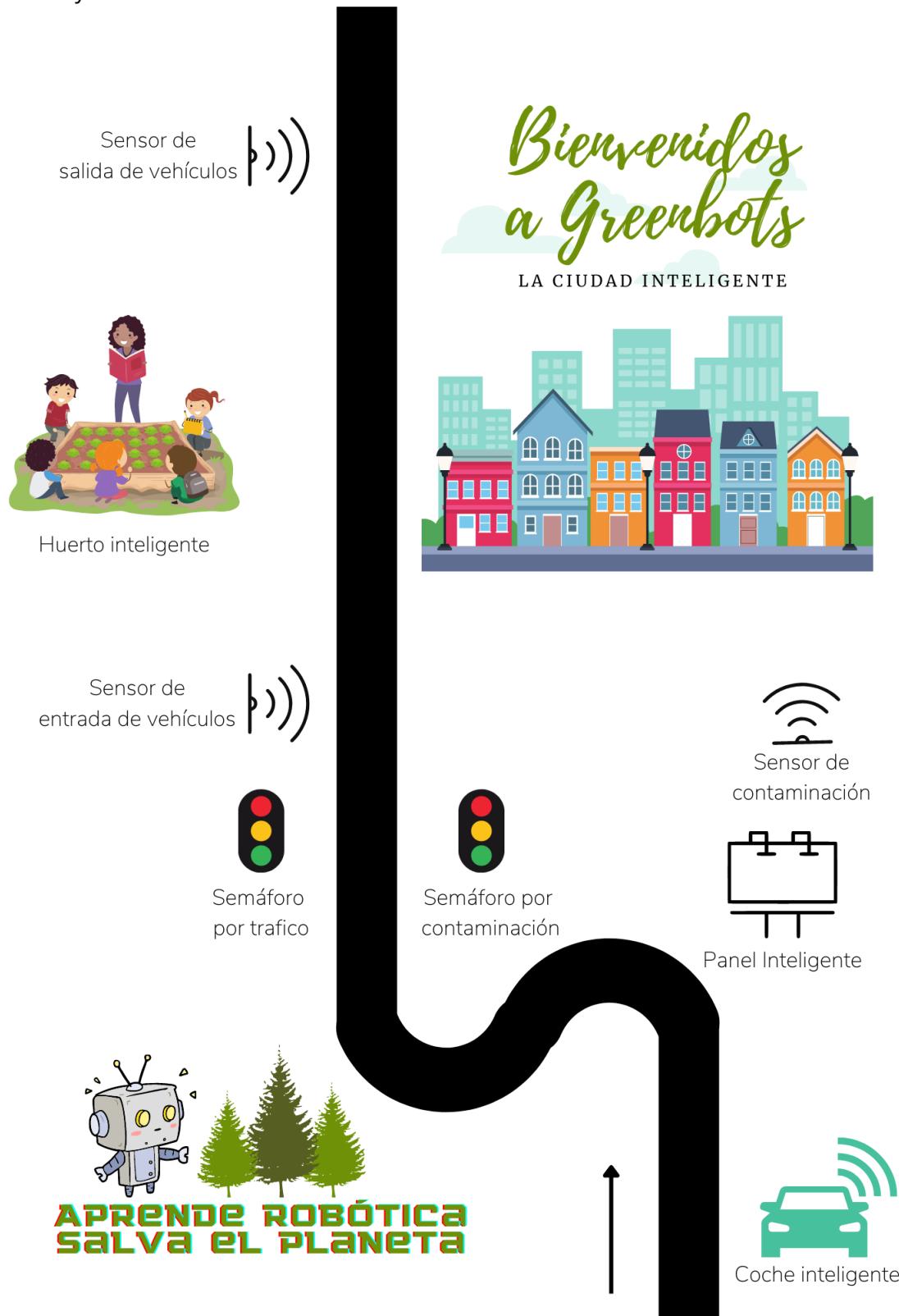
Descripción:

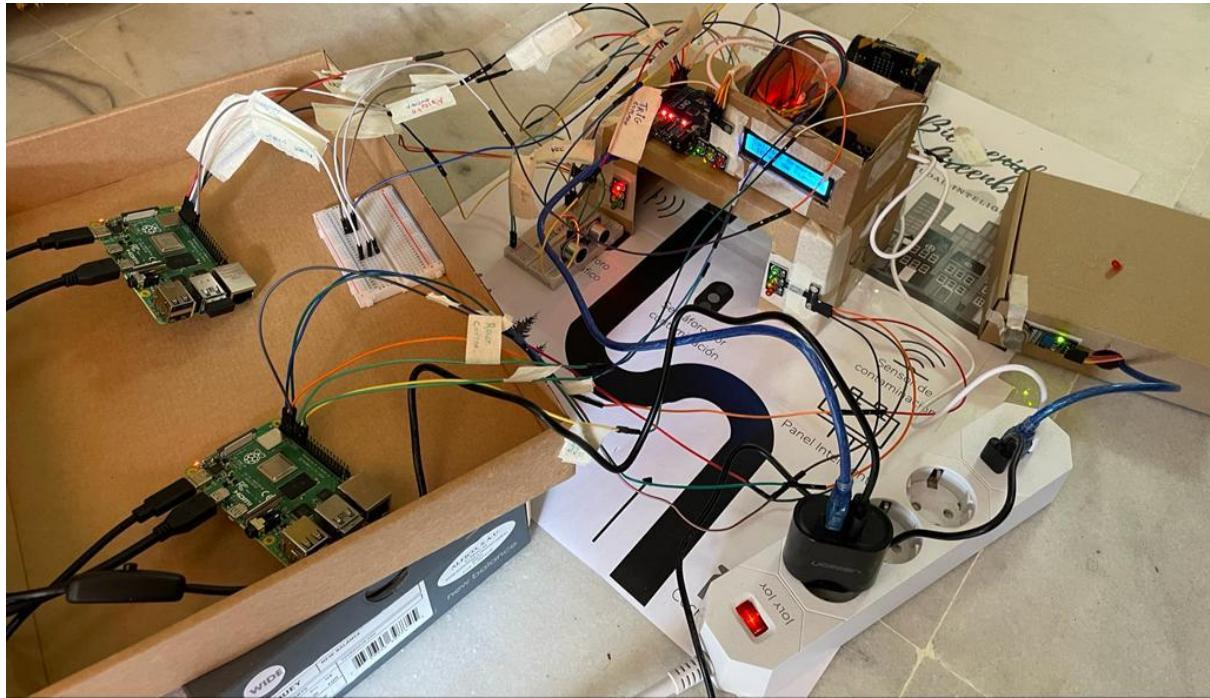
El huerto inteligente está construido con Arduino. Dispone de sensores de luz y de humedad de tierra, para activar el sistema de regado o de iluminación dependiendo de cuando las condiciones lo requieran. El sistema de regado está simulado con un LED RGB que se activa de color azul cuando está echando agua y rojo cuando no está echando agua. Cuando el nivel de luz (detectado con una fotoresistencia) es bajo, se activa la luz LED de 3W, para proporcionar luz a la planta. El código ha sido desarrollado con TinkerCAD.



4.3. Esquema final del proyecto

En el siguiente esquema pueden verse los diferentes proyectos y componentes que construyen Greenbots:





5. Conclusión

La robótica y el IoT nos han permitido construir una ciudad inteligente. Lo más novedoso de nuestro proyecto ha sido la combinación de tecnologías diferentes entre sí (Raspberry Pi, Arduino y Micro:bit) para convertir a Greenbots en la ciudad del futuro.

6. Bibliografía/Webgrafía

- <https://microbit.org/>
- <https://www.arduino.cc/>
- <https://www.tinkercad.com/>
- <https://scratch.mit.edu/>
- <https://www.canva.com/>
- <https://www.apprendiendo.es/>
- Robótica, T., 2022. *Libro Básico de Arduino*. 1st ed.