

Wireless Car Robot

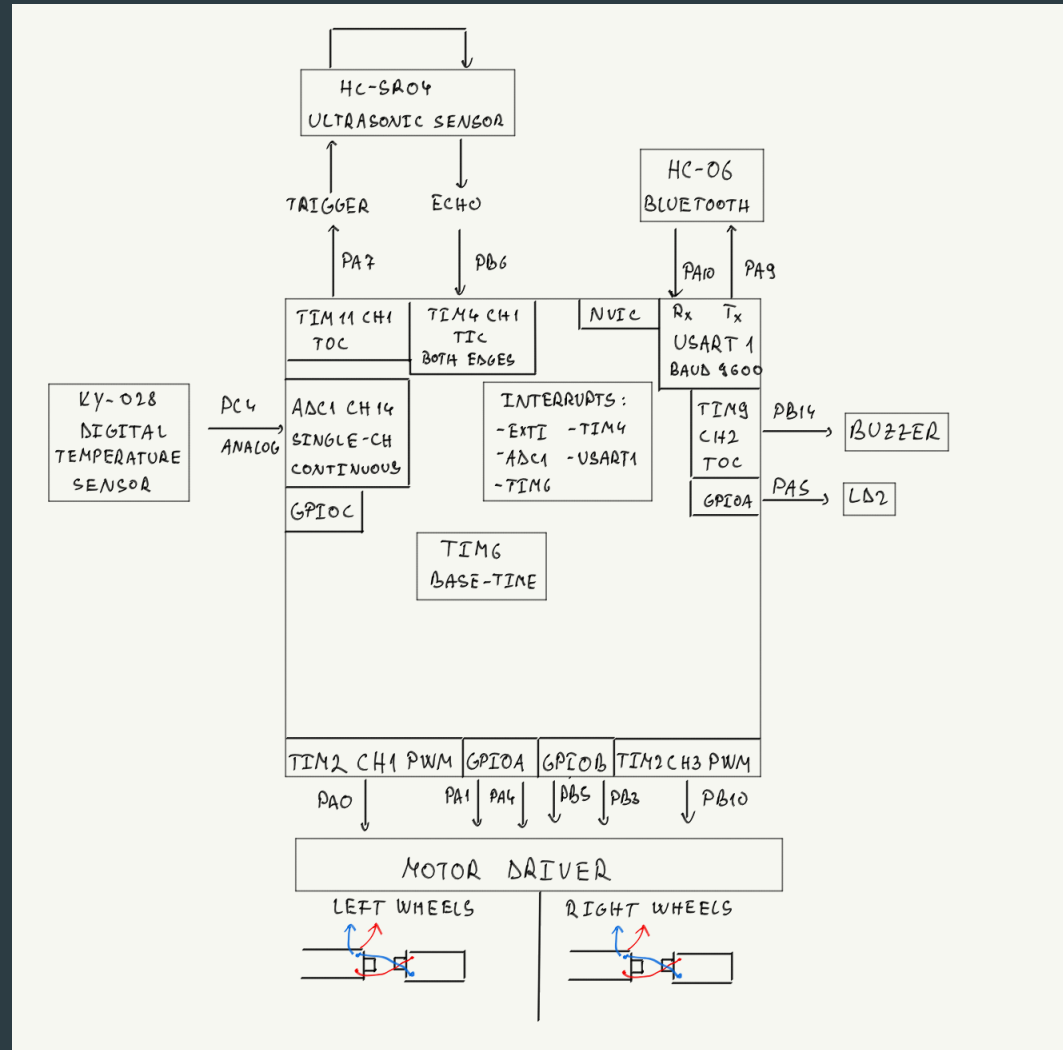
HORIA-PETRU NAZARIE and RODRIGO DURÁN MORENILLA

Components of the Robot Car

- ▶ **STM32 Microcontroller**
The brain of the robot car, responsible for processing input from sensors and sending control signals to motors and other components.
- ▶ **KY-028 Sensor**
Used for detecting temperature, this sensor sends analog data to the microcontroller for processing and decision-making.
- ▶ **DC Motors**
These motors enable the movement of the robot car, and are controlled via PWM signals for precise speed and direction adjustments.
- ▶ **Bluetooth Module**
This component facilitates wireless communication between the robot car and a Python GUI, allowing for remote control and data exchange.
- ▶ **Power Supply**
Provides the necessary voltage and current for all components, ensuring stable operation and functionality of the robot car.
- ▶ **GUI (Python)**
The graphical user interface developed in Python serves as the interactive platform for users to control and monitor the robot car's functions.

STM32 Block DIAGRAM

Detailed Analysis of STM32 Components and Their Connections



Key Components of the System

- ▶ ADC (Analog to Digital Converter)

Reads analog values from the KY-028 sensor, which is crucial for accurate temperature detection.

- ▶ GPIO (General Purpose Input/Output)

Manages motor direction and various I/O operations, allowing for versatile control in the system.

- ▶ UART (Universal Asynchronous Receiver-Transmitter)

Facilitates communication with the Python GUI, enabling seamless command and data exchange between components.

- ▶ PWM (Pulse Width Modulation)

Controls motor speed and direction by adjusting the duty cycle, providing precise control over motor operations.

The ADC (KY-028 Sensor Integration)

- Modes Selected:

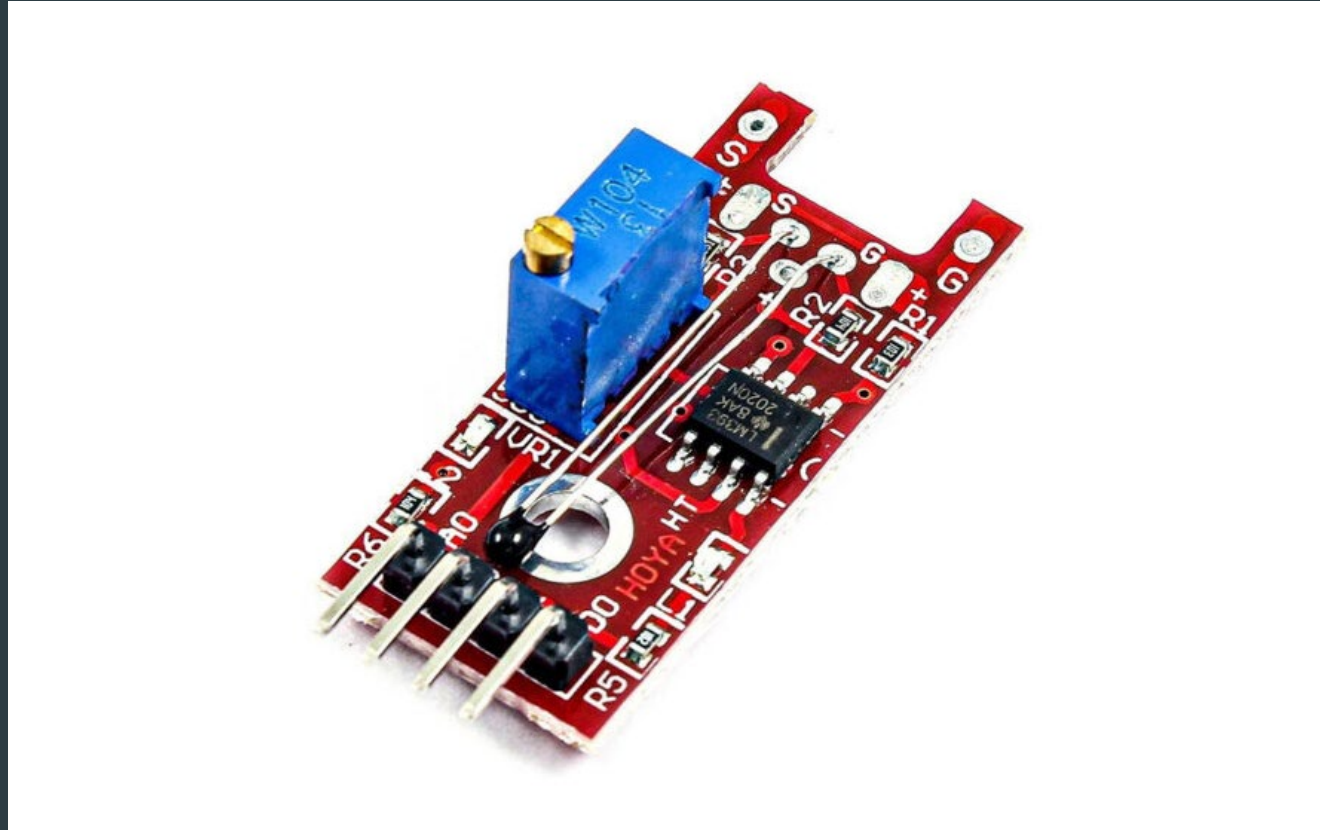
Utilizing Continuous Conversion Mode allows for real-time monitoring of temperature, while the 12-bit resolution ensures high precision with minimal error.

- Calibration:

Calibration against known temperature references guarantees the accuracy of the KY-028 readings, essential for reliable data interpretation.

- Data Processing:

The KY-028 outputs analog temperature data, which the ADC converts for digital processing. This processed data is critical for monitoring ambient conditions and optimizing motor operations.



KY-028 Digital Temperature

Explanation of the PWM (Motor Control)

- ▶ **Timer Setup:**

Configured timers to generate PWM signals, which are essential for precise motor control operations.

- ▶ **Duty Cycle Adjustment:**

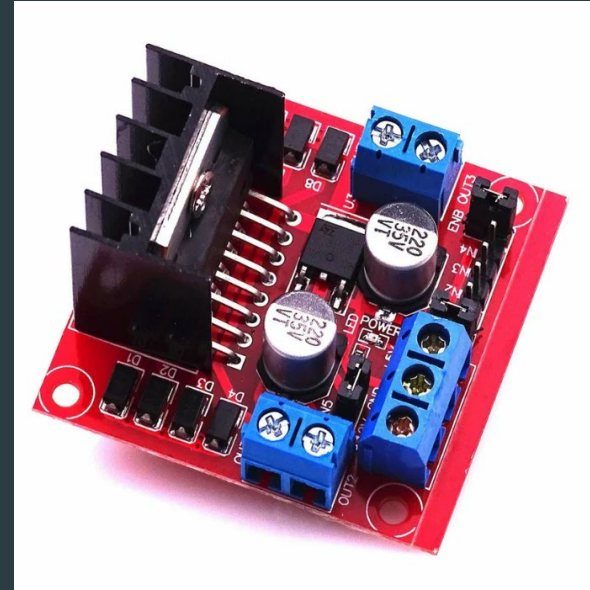
Dynamically adjusted the duty cycle to effectively control the speed of the motor, allowing for flexibility in operation.

- ▶ **Speed and Direction Control:**

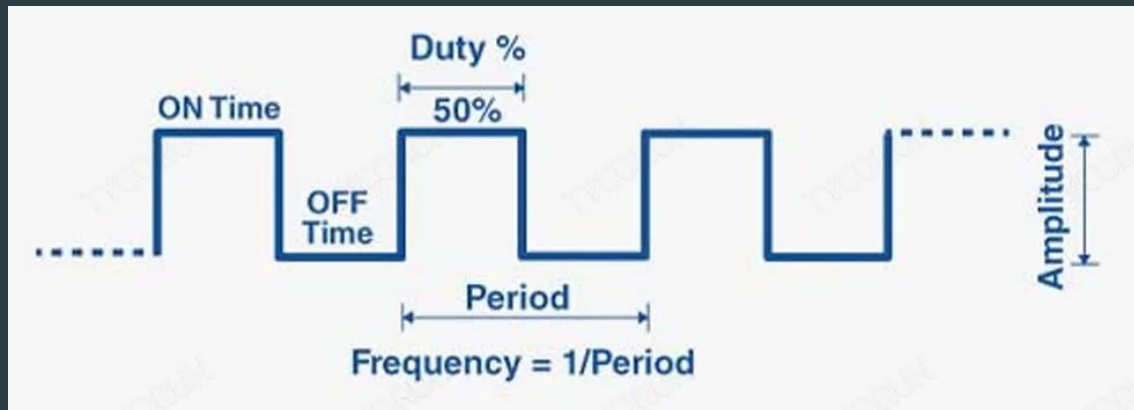
Utilized PWM signals to vary the voltage supplied to motors, enabling accurate control over speed.



DC Motor



L298N Motor Driver



The UART (Bluetooth Communication)

► Baud Rate Configuration

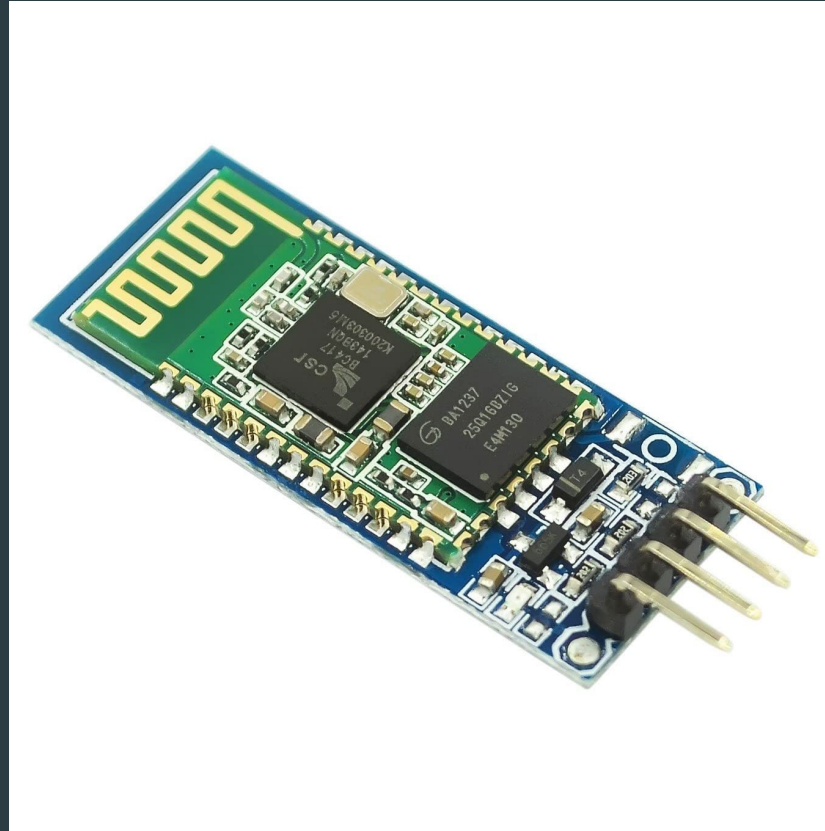
Set at 9600 bps for reliable data transmission, ensuring effective communication between devices.

► Communication Mode

Utilized interrupt mode for efficient data handling, minimizing CPU load and enhancing performance.

► Process Overview

STM32 microcontroller sends and receives data via the Python GUI, allowing users to control the car remotely with ease.



HC-06 Bluetooth Module

Explanation of the code

- ▶ EXTI Peripheral:
 - ▶ When the board button is pressed, it generates an interrupt, which acts as a trigger to initialize all other peripherals.
- ▶ Bluetooth Module:
 - ▶ The HC-06 Bluetooth module communicates with the STM32 via UART.
 - ▶ It receives commands (motor controls) and sends feedback (sensor data) to the controlling device.
- ▶ DC Motor Control (PWM Output):
 - ▶ The STM32 generates PWM signals on specific GPIO pins to control the speed and direction of the DC motors.
 - ▶ The PWM duty cycle is adjusted based on ultrasonic sensor data

Explanation of the code

- ▶ Buzzer Control (Digital Output):
 - ▶ The buzzer is activated via a GPIO pin when certain conditions are met (obstacle detection).
- ▶ Ultrasonic Sensor
 - ▶ The microcontroller uses a timer to send trigger pulses to the ultrasonic sensor.
 - ▶ It measures the echo pulse duration to calculate distance, using timer peripherals with interrupts.
- ▶ ADC Peripheral:
 - ▶ Configured in single-channel, continuous mode to read sensor inputs like temperature

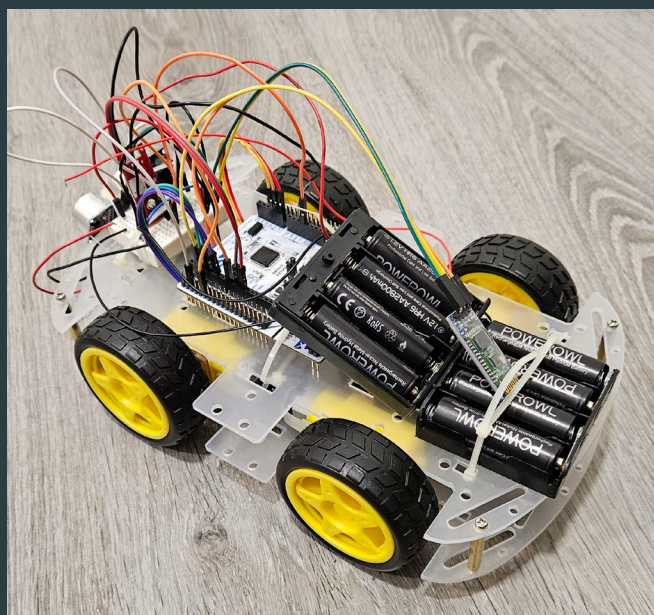
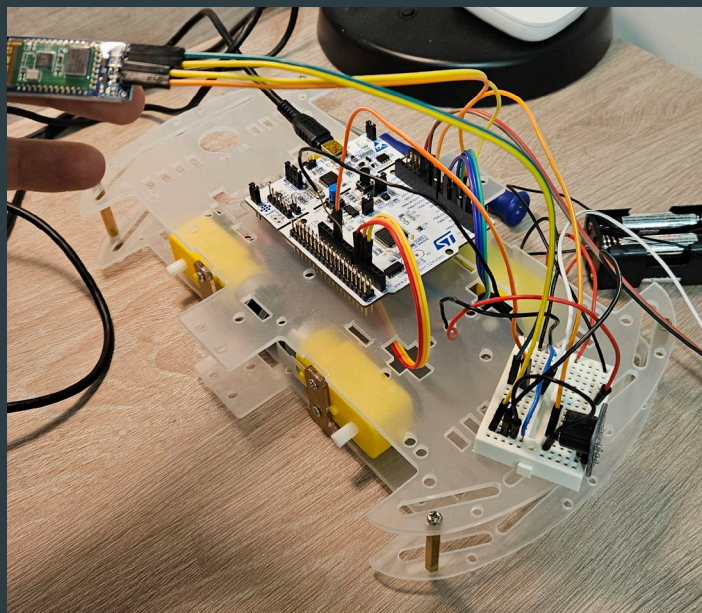
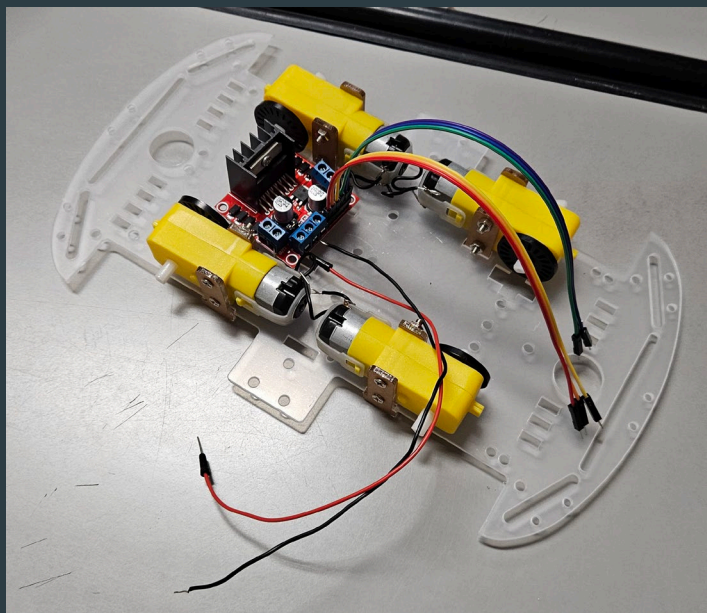
Challenges and Solutions

- ▶ Transmitting Data

Sending temperature sensor data to Bluetooth module.

- ▶ Communication Interference

Interference in Bluetooth communication using the USART serial communication protocol.



Thank you!