IOT RC CAR

The RC car system is visualized by the following diagram:

Afbeelding met tekst, diagram, schermopname, Plan

Automatisch gegenereerde beschrijving

Figure 1: Schematic of the RC CAR

The project has two subsystems. The controller subsystem and the car subsystem. Both utilize a PSOC. The subsystems communicate using Bluetooth low energy (BLE).

# Controller subsystem

The controller subsystem utilizes a PSoC 6 CY8CKIT-062-WIFI-BT. This subsystem controls the movement of the car. The input for this is a single joystick that is read out by the PSoC using the on board 12-bit ADC. The subsystem also has two seven segment displays to show the speed of the car. The joystick readout is sent via BLE to the car subsystem. The BLE communication is bidirectional, so the controller also receives the speed of the car to display on the seven-segment display. The PSoC uses RTOS to schedule the different tasks.

# Car subsystem

The car subsystem uses the CY8PROTO-062-4343W PSoC 6. This PSoC is connected to the controller via BLE. It is connected to a MPU6050 breakout board which has an accelerometer, a gyroscope and a temperature sensor. This interfacing is done via I2C. The PSoC reads this data and pushes it to the user interface via MQTT over the HiveMQ broker. This is a secure connection. This system also uses RTOS for scheduling.

The PSoC drives the H-bridge breakout board. This board features a L293D H-bridge IC for controlling two motors. This is ideal because the car frame used in the project had two drivable DC motors.

The car subsystem is powered by a 12V rechargeable Lithium battery. The voltage was lowered to 5V using a DC/DC step down converter for powering the PSoC and H-bridge IC. The motors themselves are driven with 12V.

All of this is soldered onto a PCB to have a cleaner design than just a breadboard. This board is depicted in figure 2.

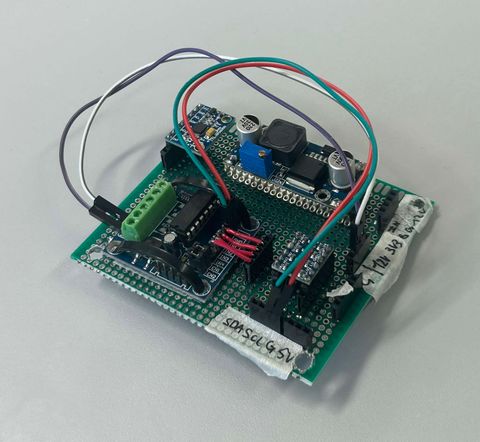


Figure 2: Car subsystem circuit board

# Server UI

The web user interface runs on a backend python flask application. The user interface displays the MPU6050 data and has a cube that rotates and simulates the orientation of the car. The UI has two commands that it can send to the car PSoC. These are one button to toggle the motors. With this the motors can be disabled via secure MQTT connection. The second is a reset of the car’s orientation. If the orientation on the UI does not match the car’s orientation, then with this reset button they can be matched again.