

V2V COMMUNICATION: EMULATING REAL CAR ACTIONS



MEMBERS

HOSSAM ELWAHSH
AHMED SAKR
AHMED HISHAM
NADA SALLOUM
SALMA FARAGALLA
NORHAN MOHAMED



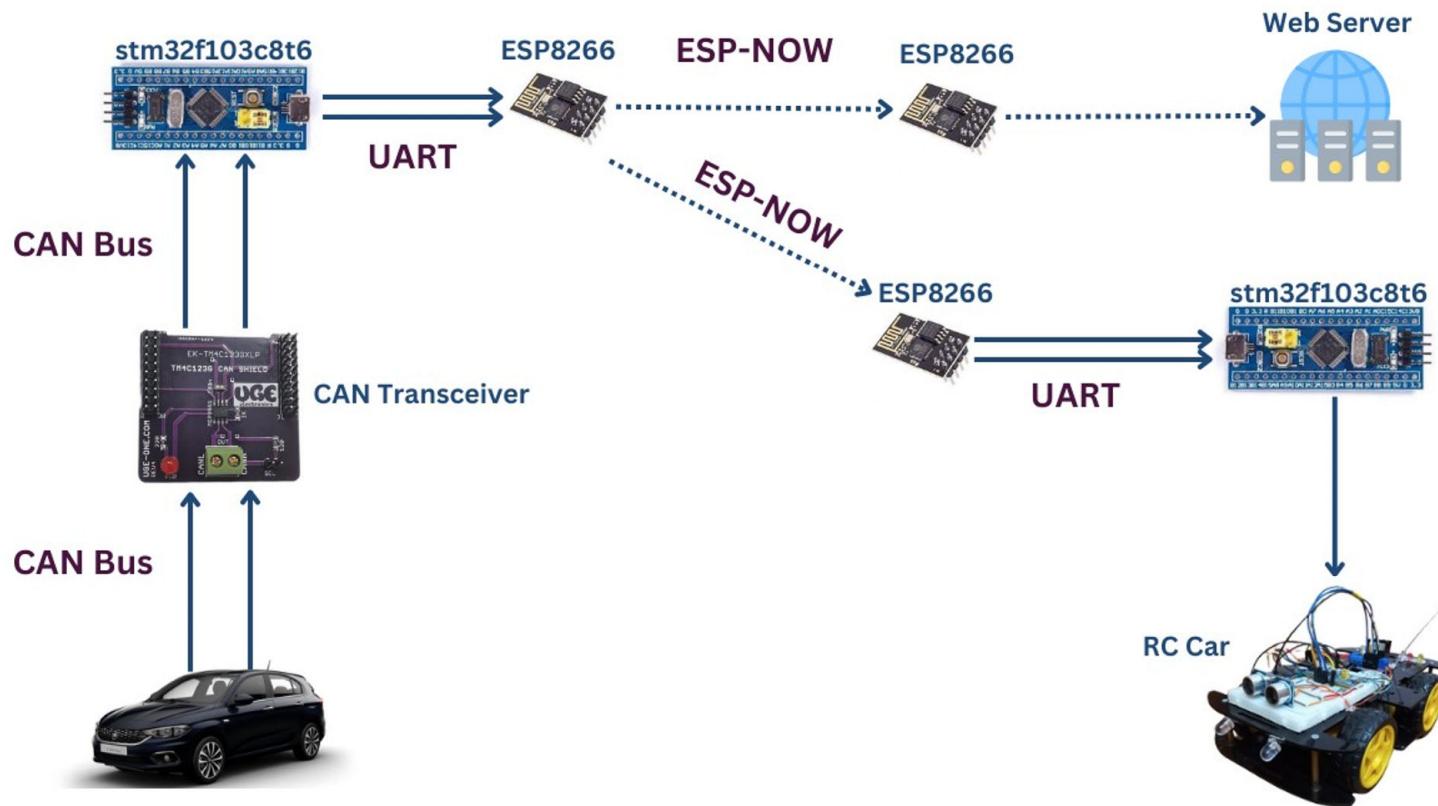
AGENDA

Project Objectives	System	CAN Data	Wireless Data Transmission	Data mapping and processing	Ultrasonic	Vehicle Control	Lights handling	Video Showcase
	System Overview Sequence Diagram	Analysis Findings	Client – server protocols ESP-NOW	RTOS Tasks	RTOS task Issues	Direction Speed Issues	RTOS Tasks Future Enhancements	

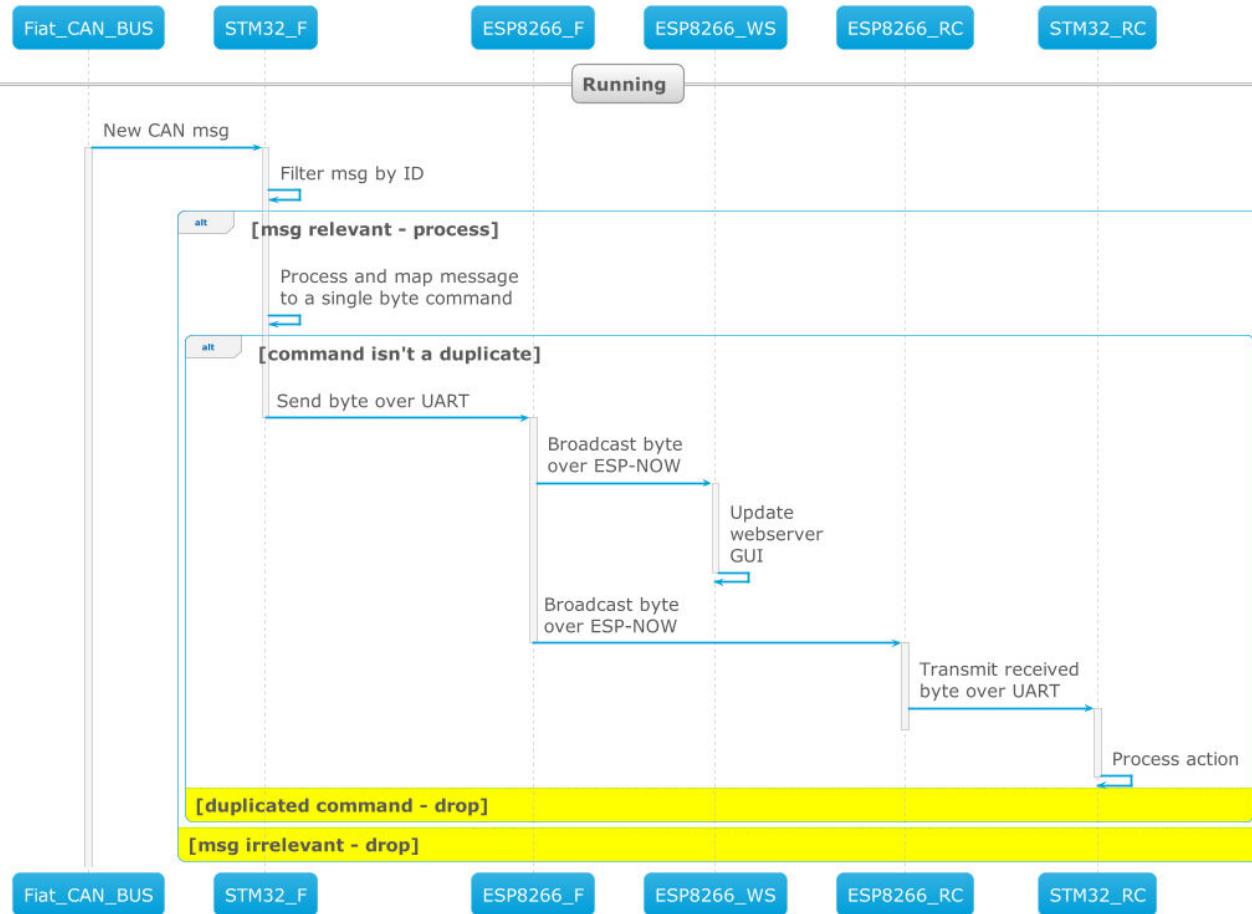
PROJECT OBJECTIVES

- Tap into the Fiat Tipo 2019 HS-CAN network to collect vehicle data
- Analyze, find and filter following data:
 - Current transmission selection (P, R, N, D)
 - Throttle position
 - Steering wheel position
 - A selection of car lighting
- Map these data to a single byte for faster transmission
- Transmit these data to an RC Car in real-time to clone and emulate the same actions
- Provide some safety features to prevent accidents
- Achieve minimal delay V2V communication

SYSTEM OVERVIEW

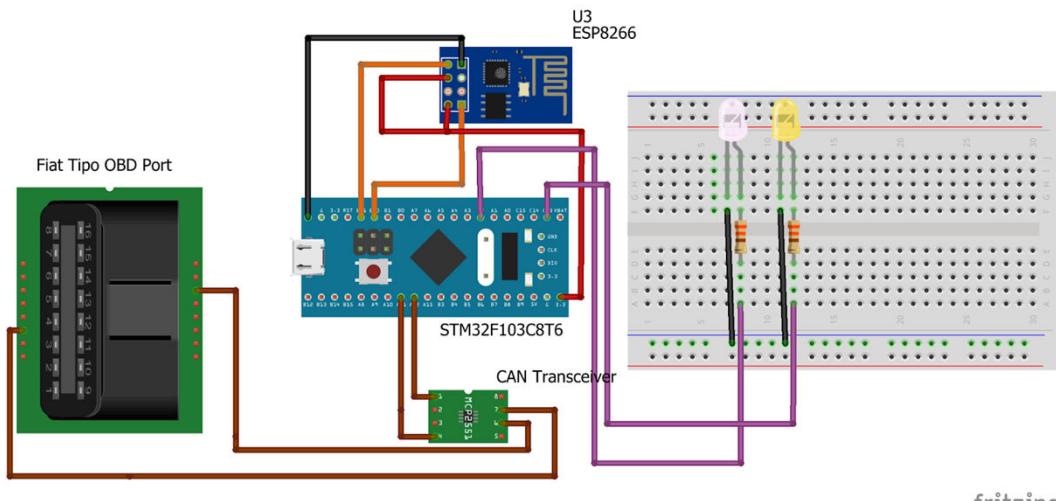


BRIEF SYSTEM SEQUENCE DIAGRAM

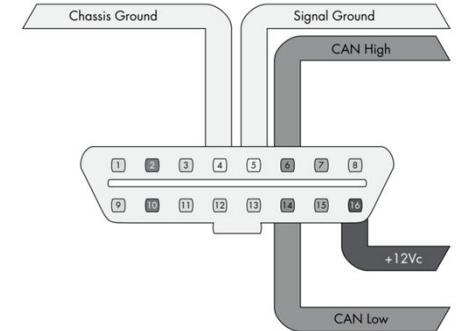


ANALYSIS

- RPi 4 Model B
- MCP 2515 CAN Module
- Cansniffer library

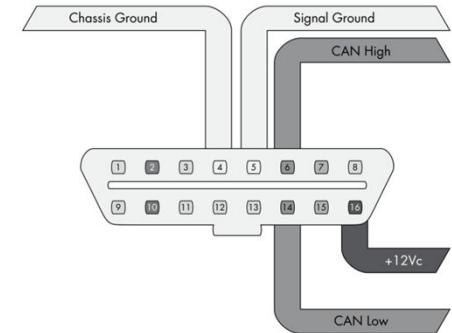


fritzing

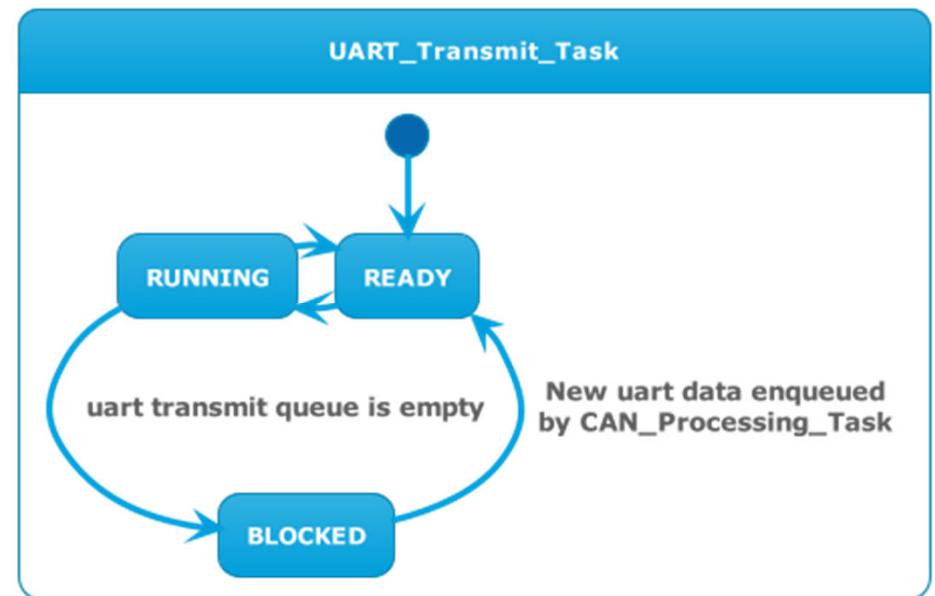
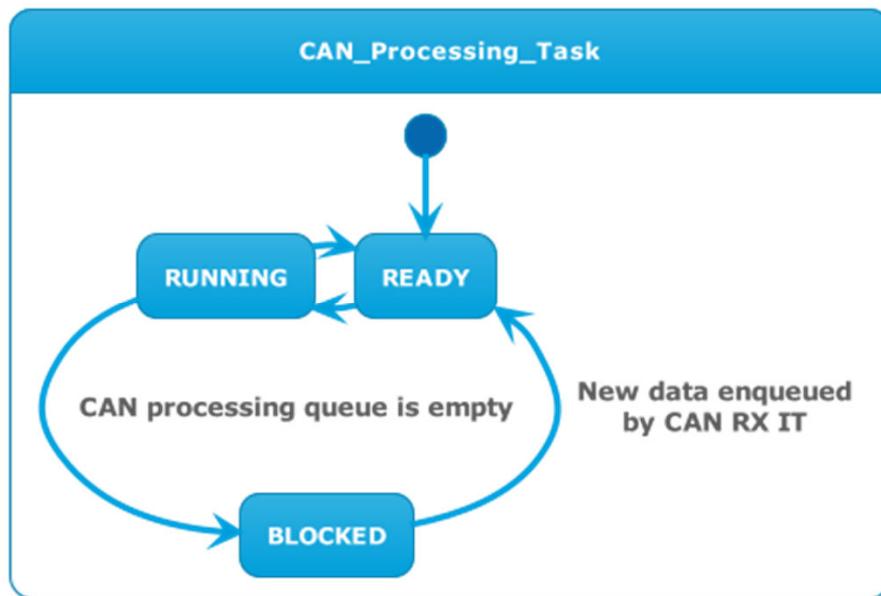


FINDINGS

Item	CAN Msg ID (hex)	Relevant Frame Bytes	Mapped Header	Mapped Data Range
Throttle	0x1F0	(00 00) XX XX .. XX (1F 40) XX XX .. XX	0x1(_)	0x0 – 0x4
Steering	0x0DE	(2C) XX XX XX .. XX (09) XX XX XX .. XX	0xE(_)	0x0, 2, 3, 8, C
Transmission	0x2EF	(FF) XX XX XX .. XX (1F) XX XX XX .. XX	0x7(_)	0x0, 1, 7, F
Lights	0x73E	Almost Full Frame	Multiple	Multiple

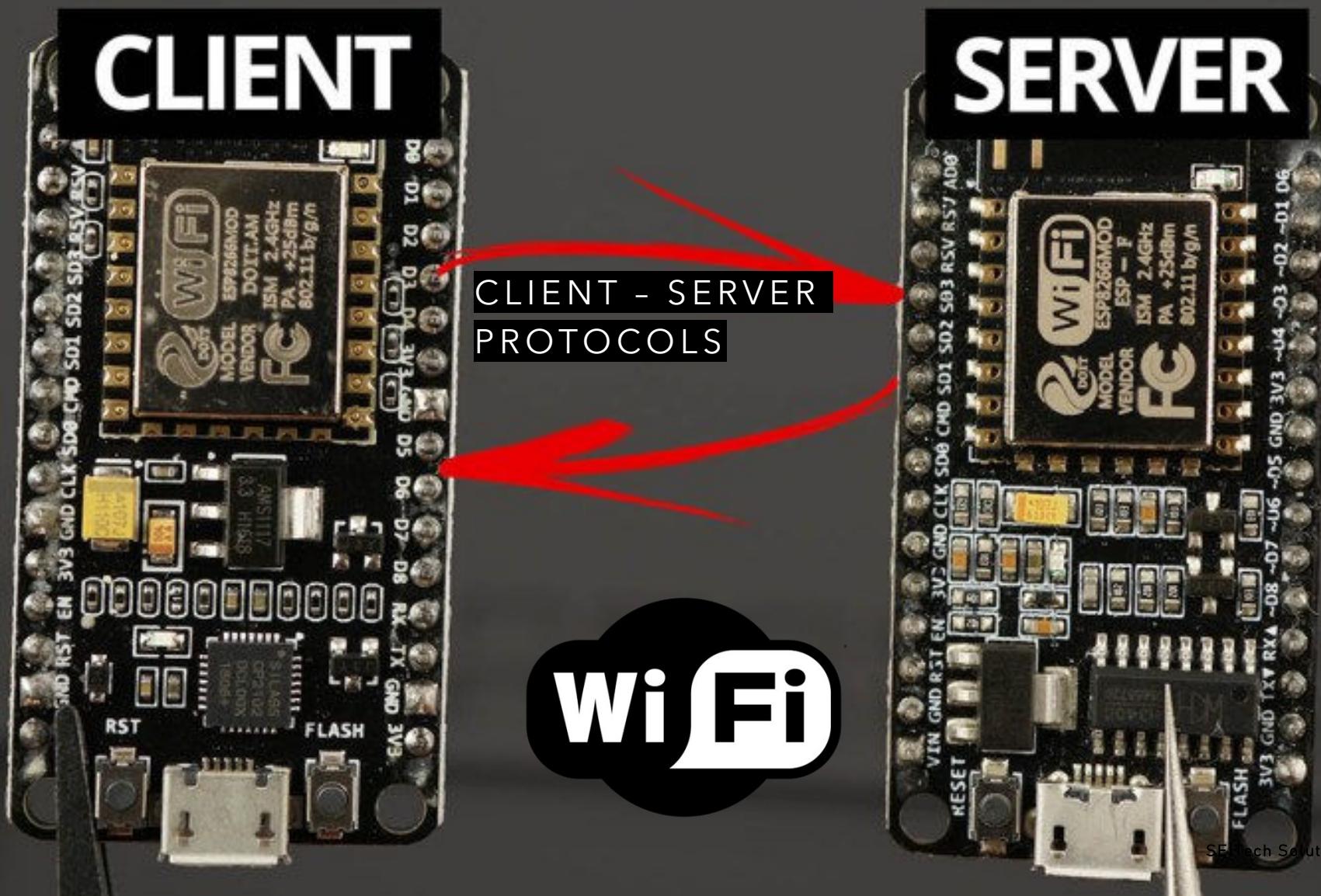


RTOS TASKS



DATA TRANSMISSION

- Using ESP8266



PROS AND CONS

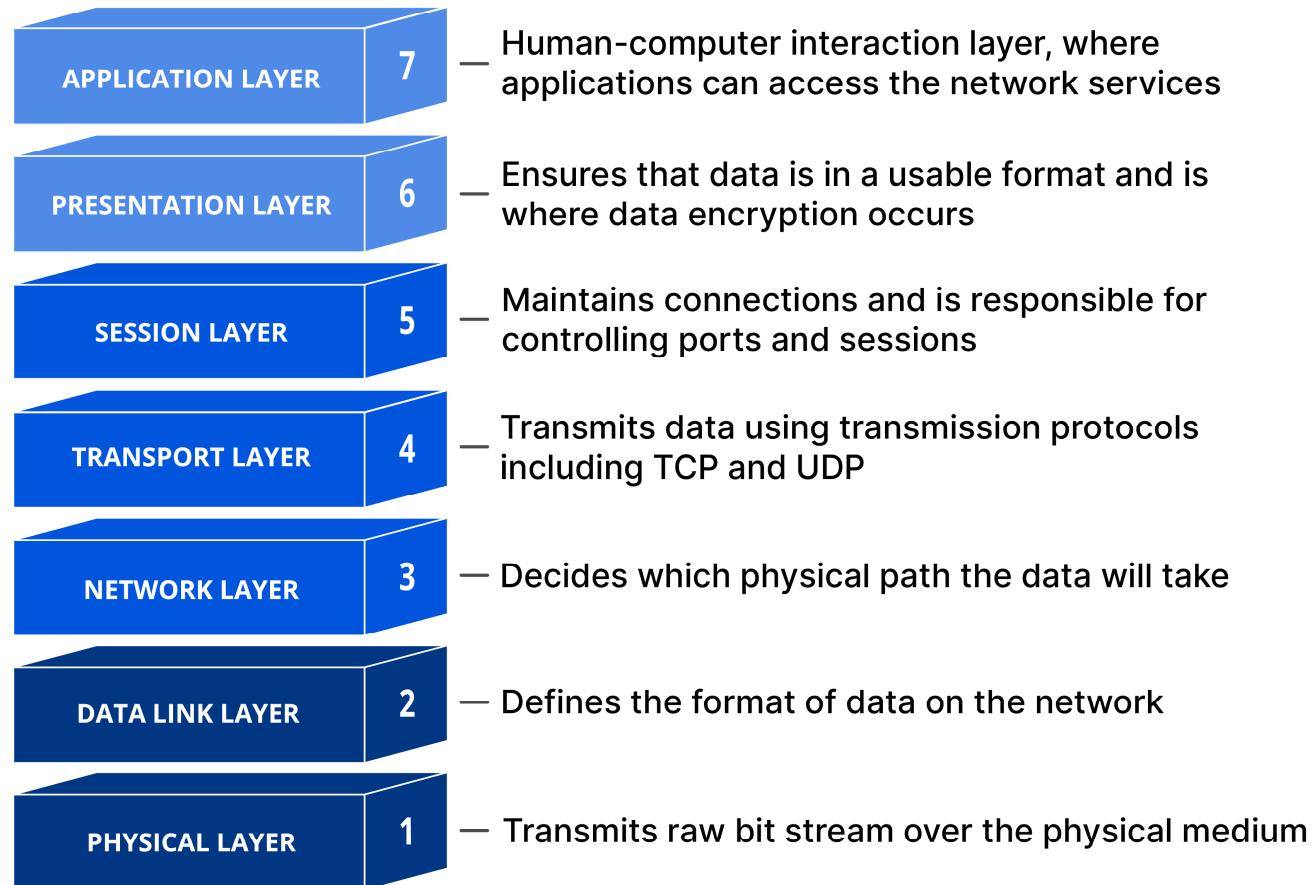
PROS

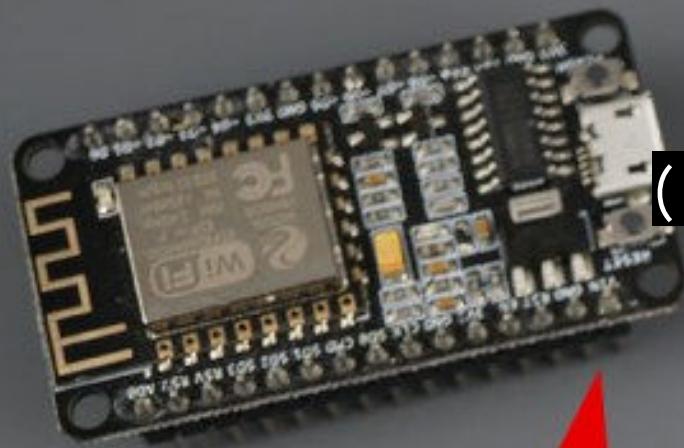
- Wireless
- Can be monitored
- User can interact with

CONS

- Large packet size
- High memory and power consumption
- Can be spoofed by a hacker

OSI model

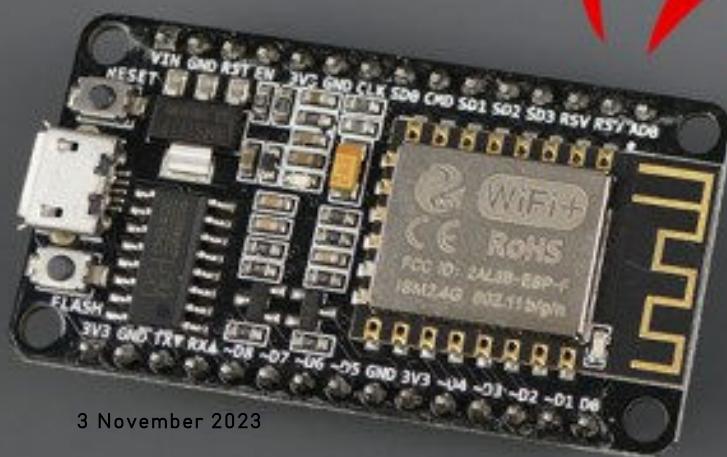




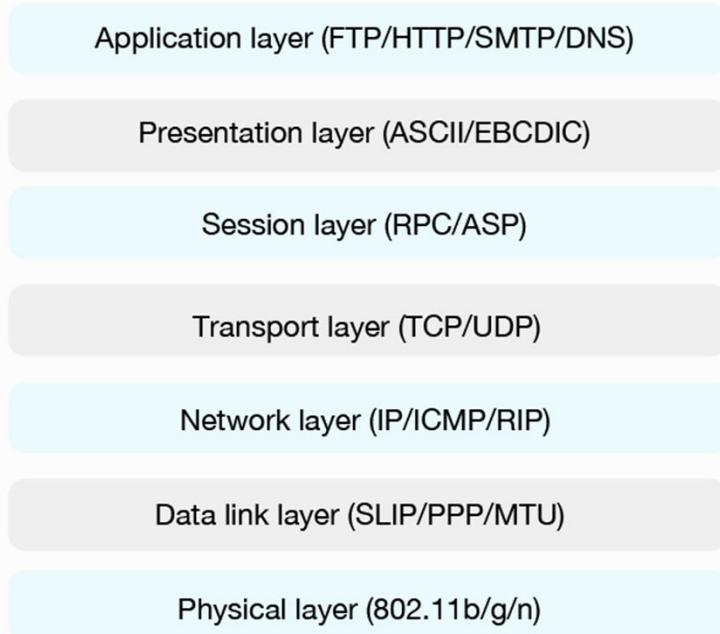
ESP-NOW
(ONE-TO-MANY)



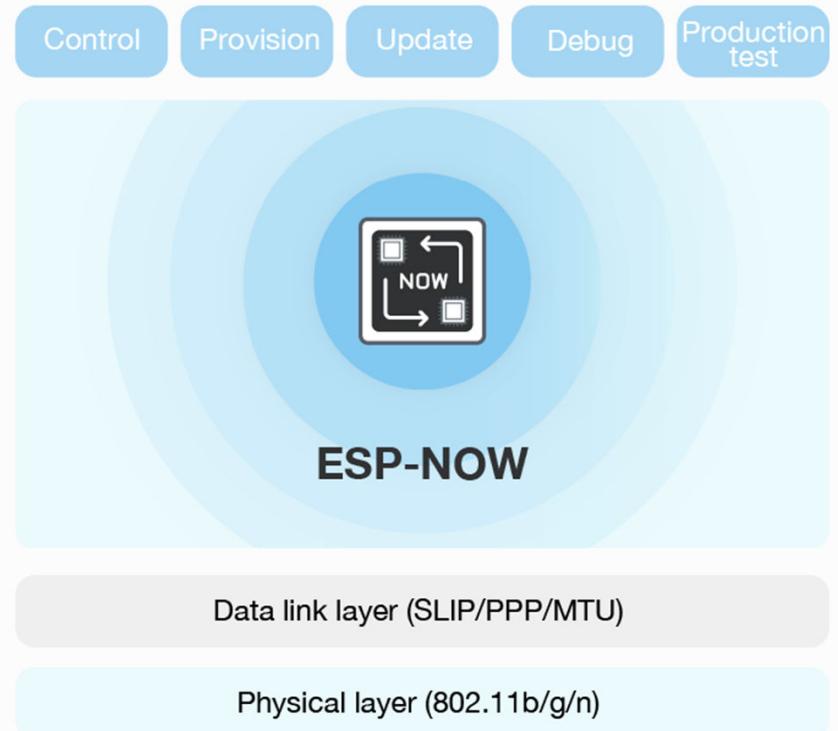
ESP NOW

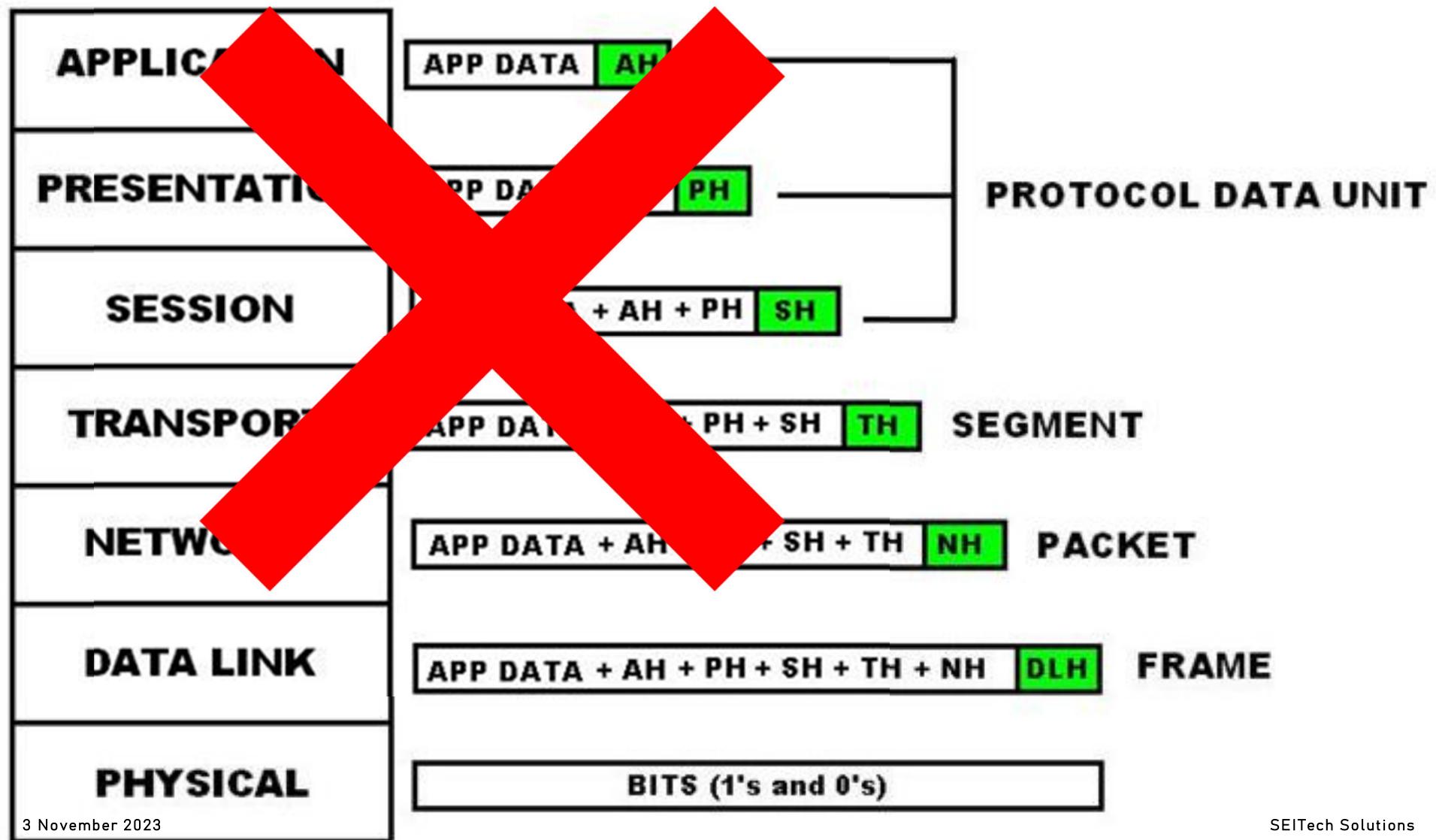


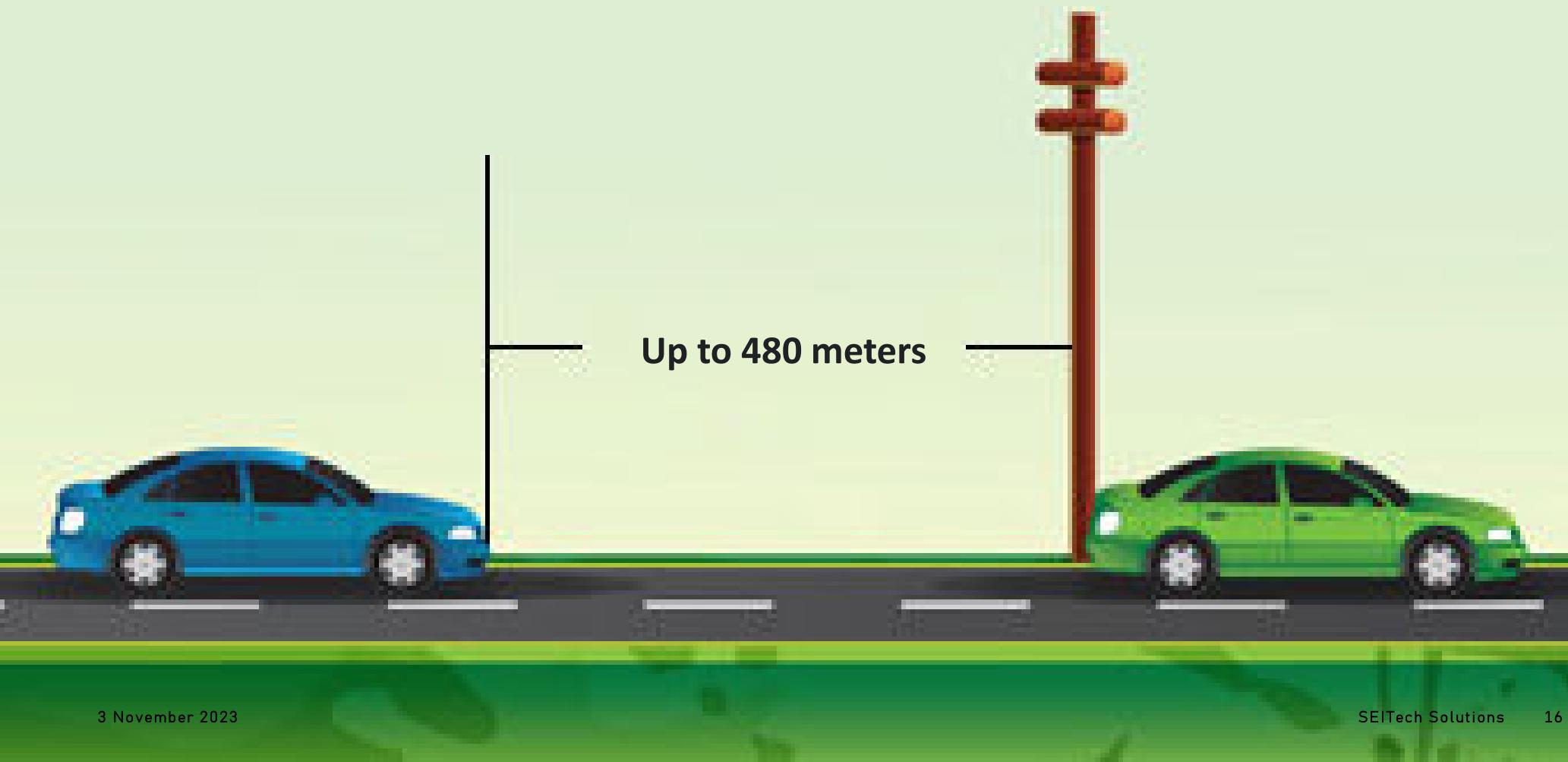
OSI Model



ESP-NOW Model





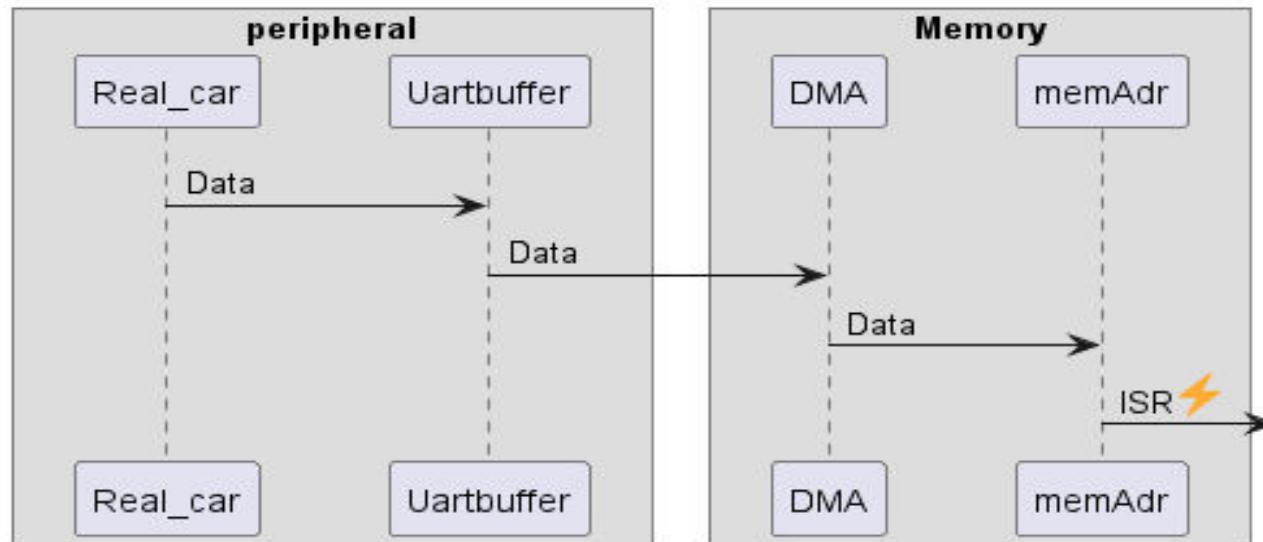


RECEIVE INTERRUPT

HOW DOES IT WORK?

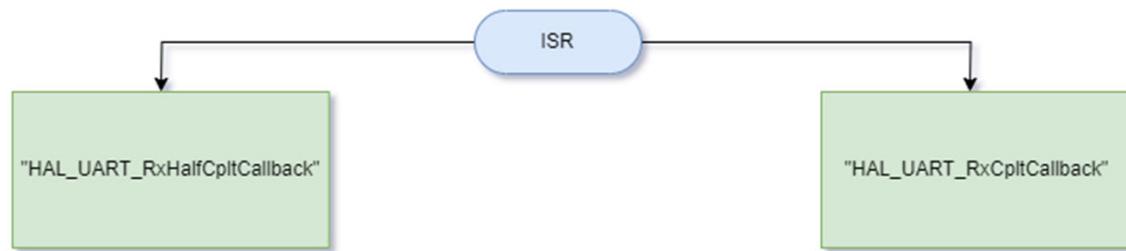
DMA: Delivers the received data from UART Buffer to Memory.

ISR: Fired once the data is being transferred.



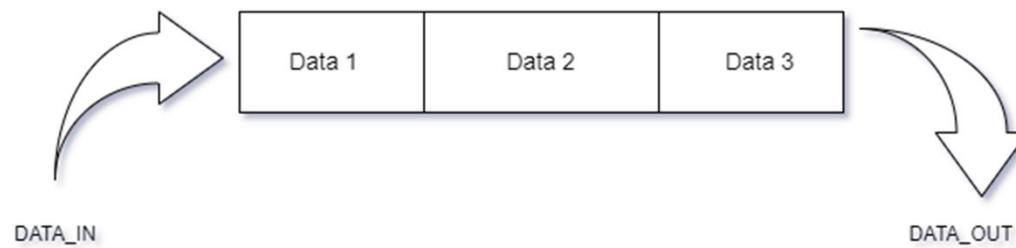
TWO TYPES OF ISR

- The interrupt "HAL_UART_RxHalfCpltCallback" fires at half of the buffer size.
- The interrupt "HAL_UART_RxCpltCallback" fires at the end of the buffer size.



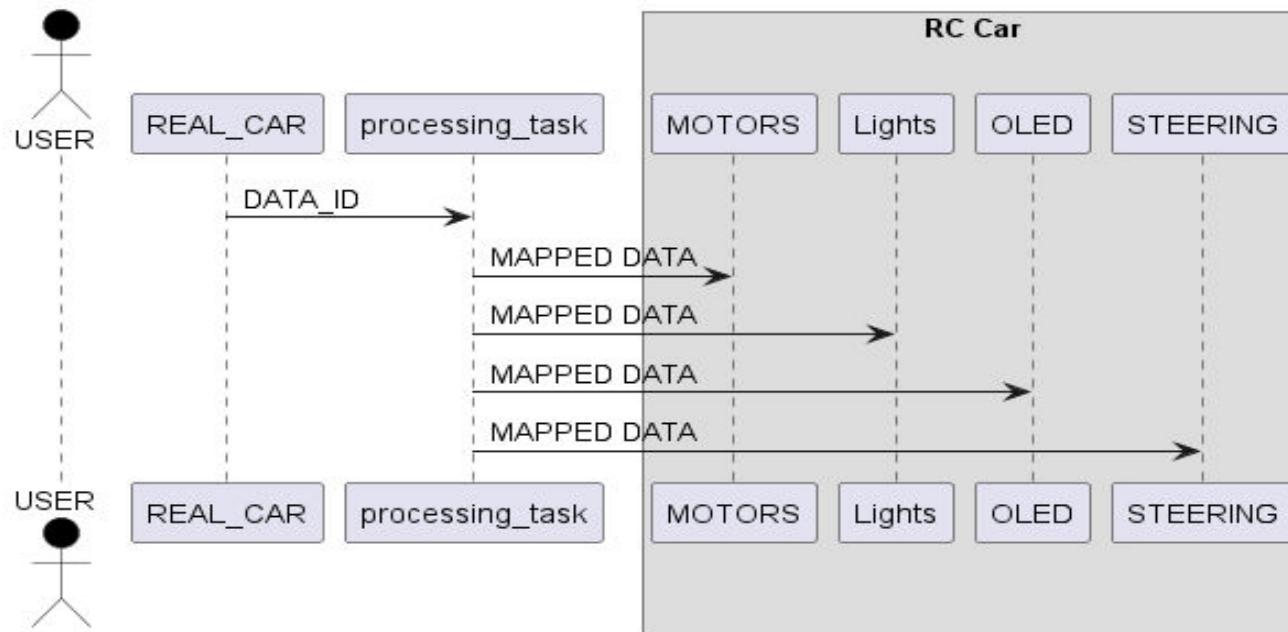
WHY USING QUEUES?

- Preventing losing the data when receiving new data while processing previous one.

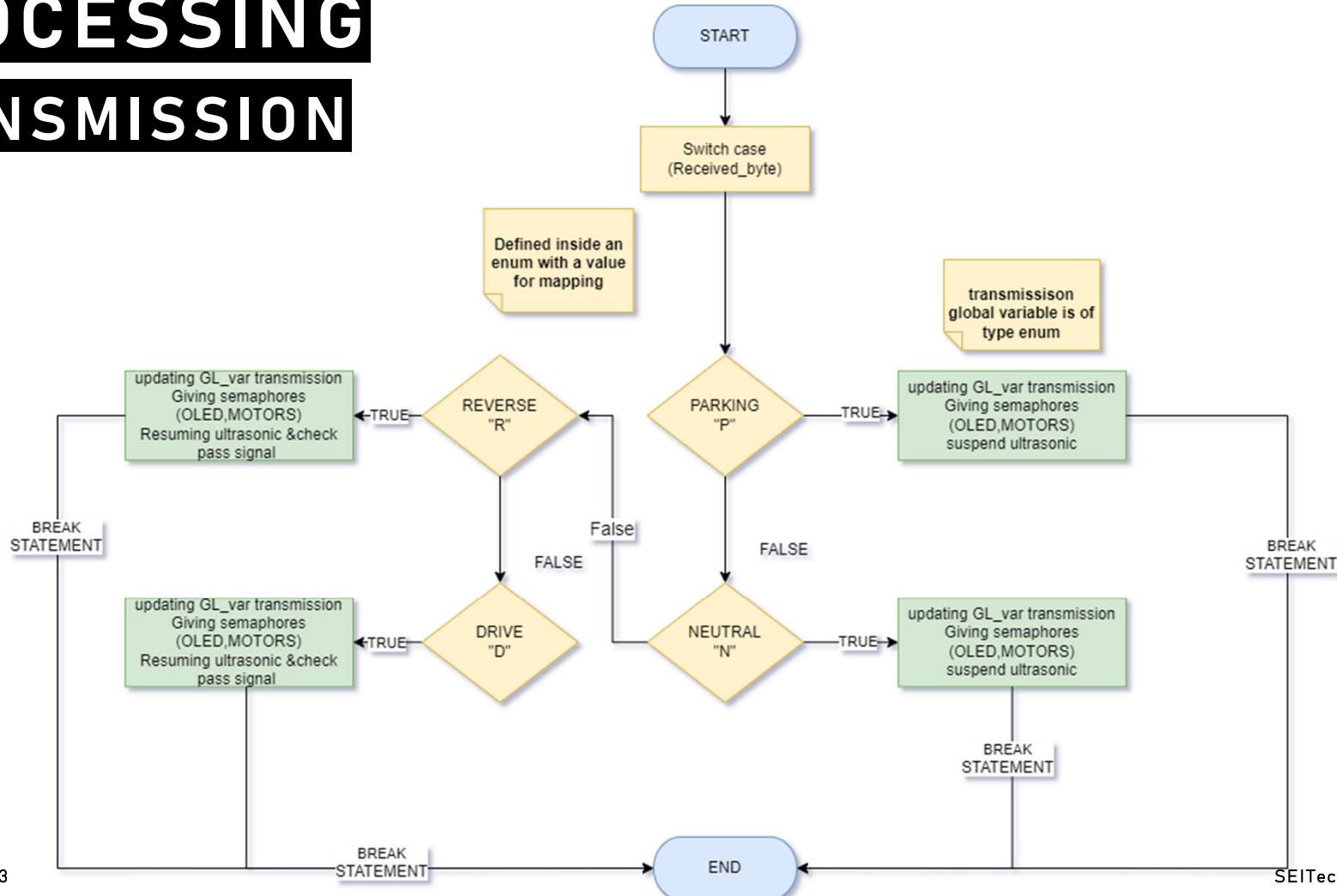


PROCESSING TASK

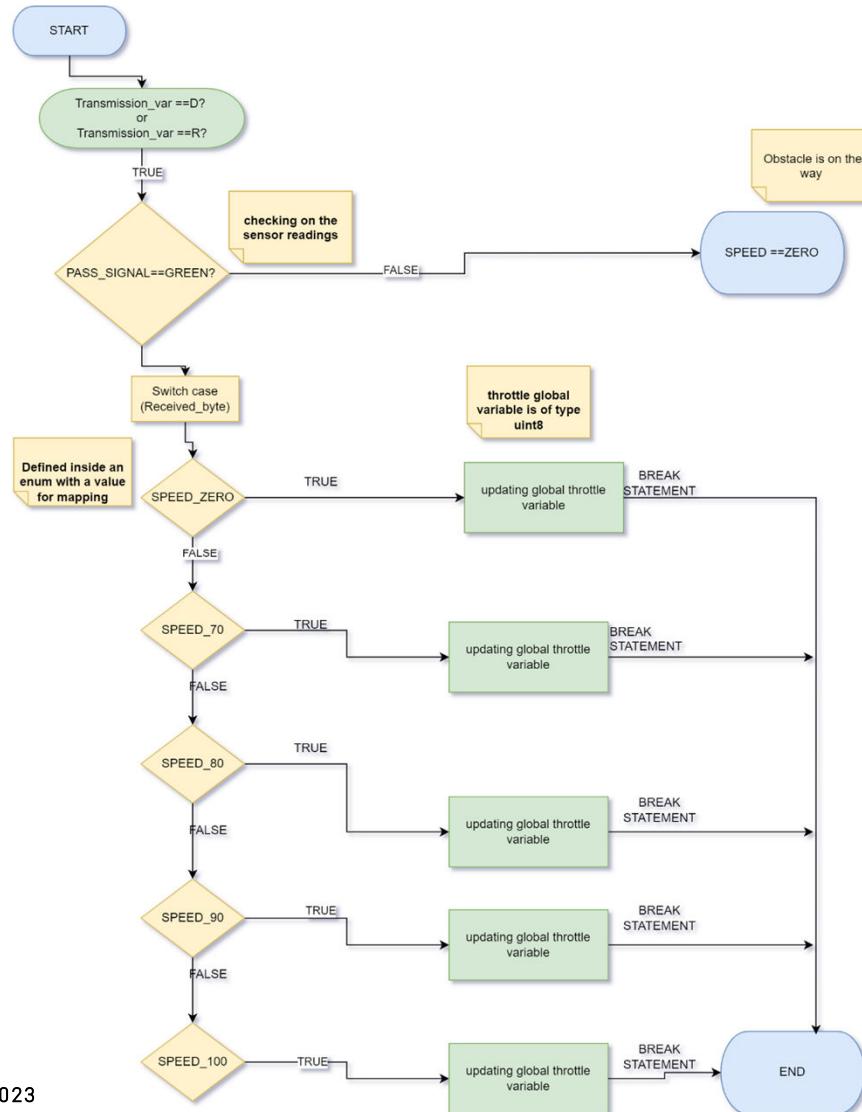
- Responsible for mapping all the data coming from the real car through esp8266.

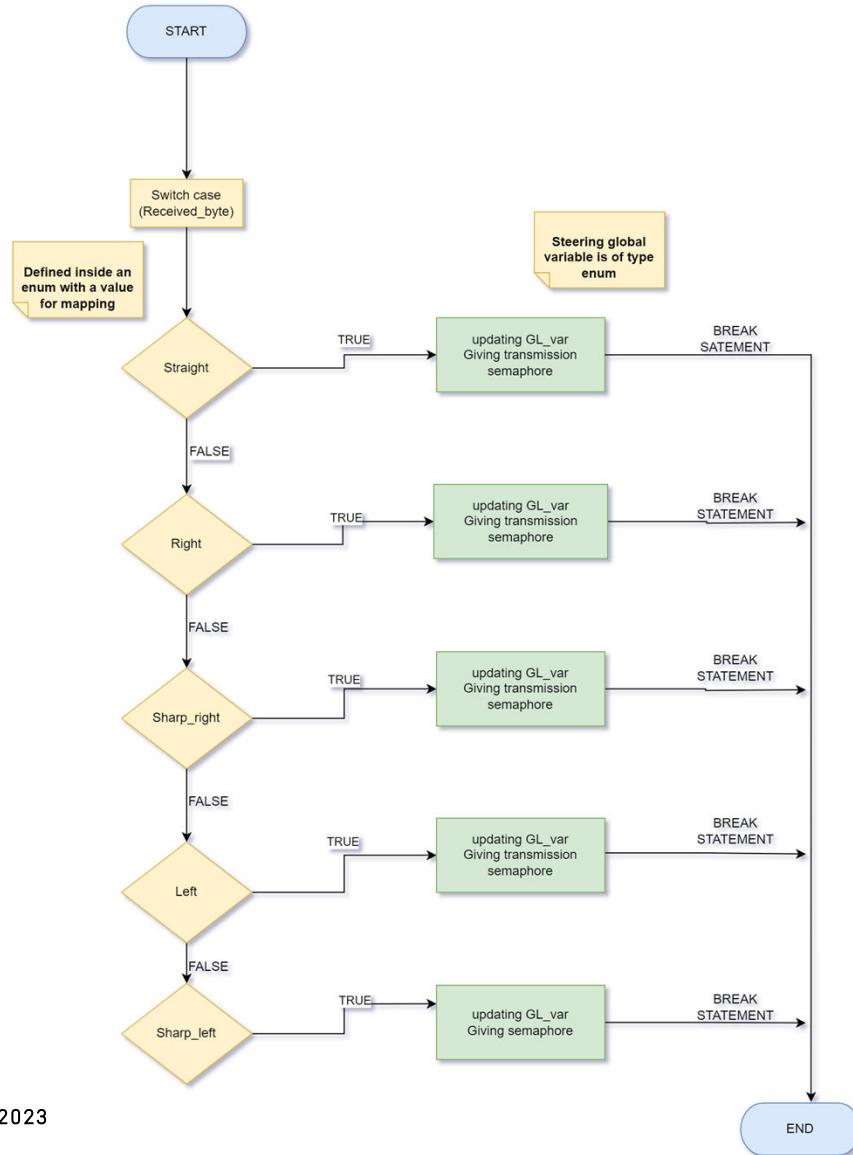


PROCESSING TRANSMISSION



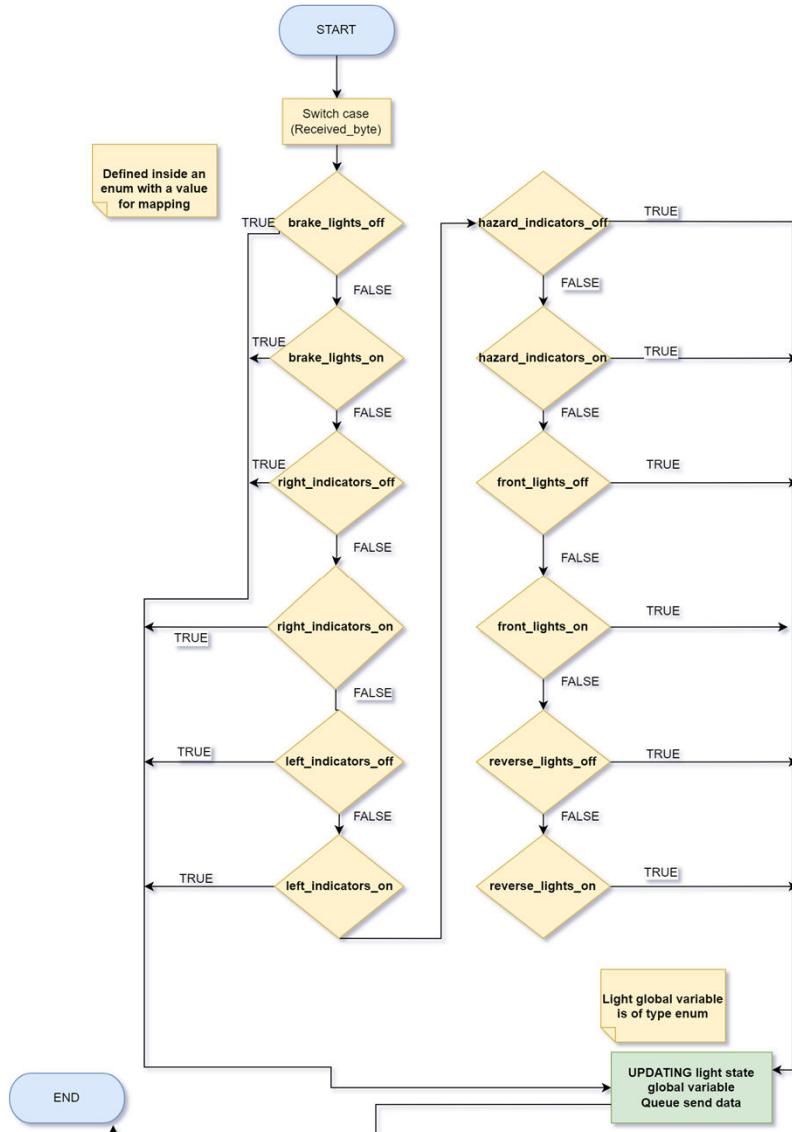
PROCESSING THROTTLE





PROCESSING STEERING

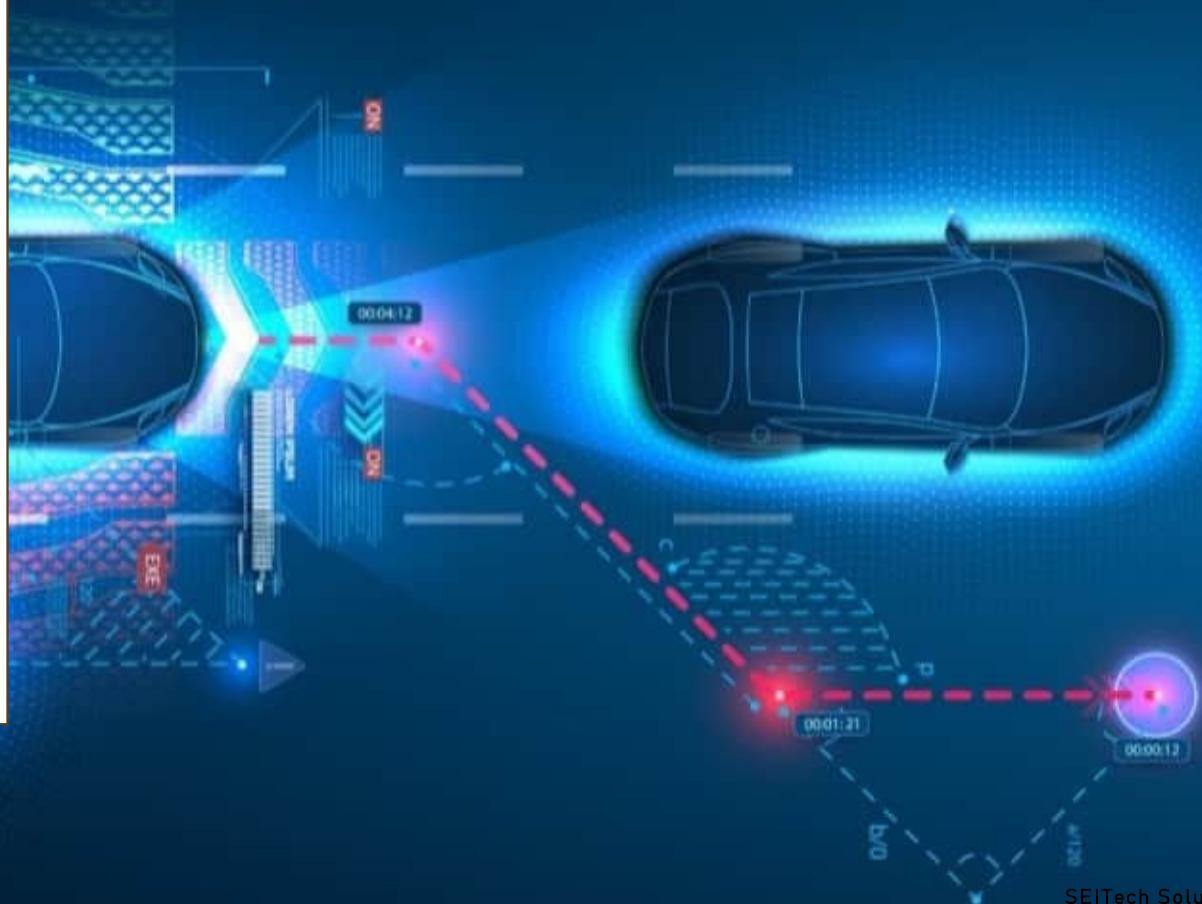
PROCESSING LIGHTS



ADAS SYSTEM

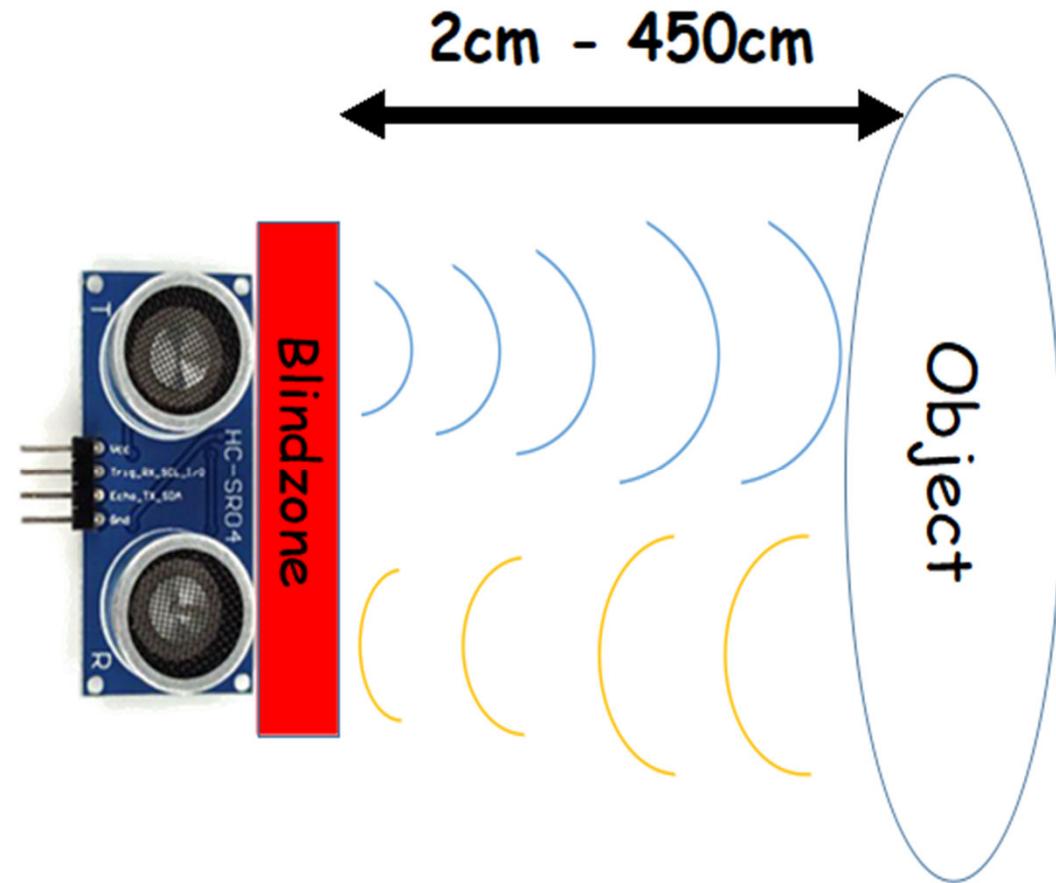
Content :

- Why we need it
- Type of ultrasonic we used
- The sequence of the system
- Problems

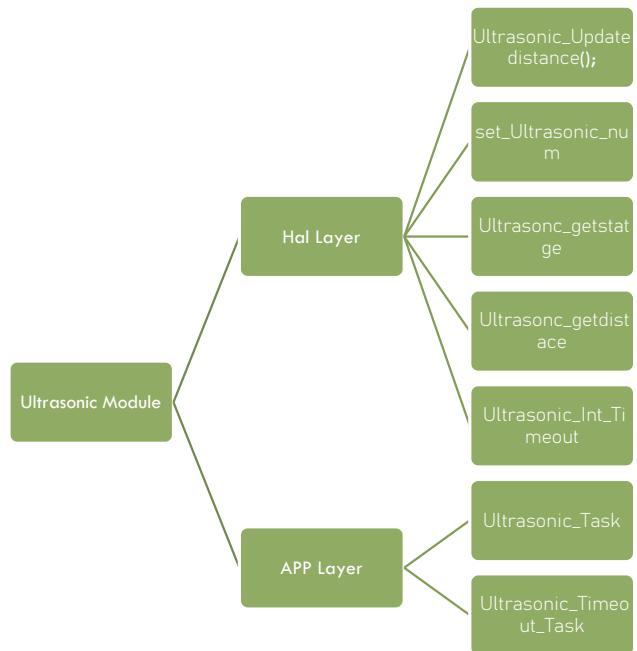


ULTRASONIC SENSOR (HC-SR04)

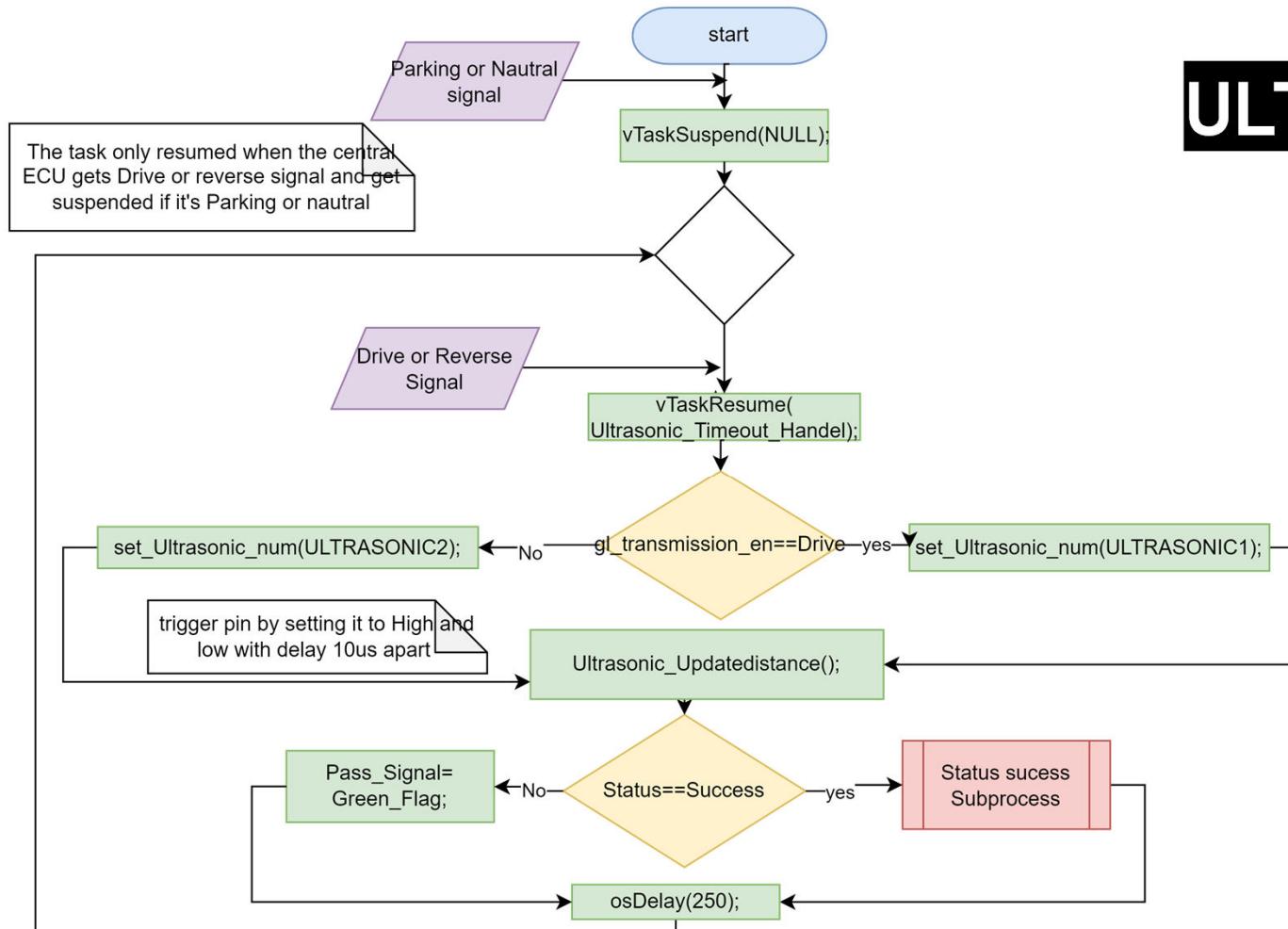
OPERATING VOLTAGE	5VDC
Operating Current	15mA
Max. Range	4.5m
Nearest Range	2cm
Input Trigger Signal	10us min. TTL pulse
Speed of Sound in air	340m/sec



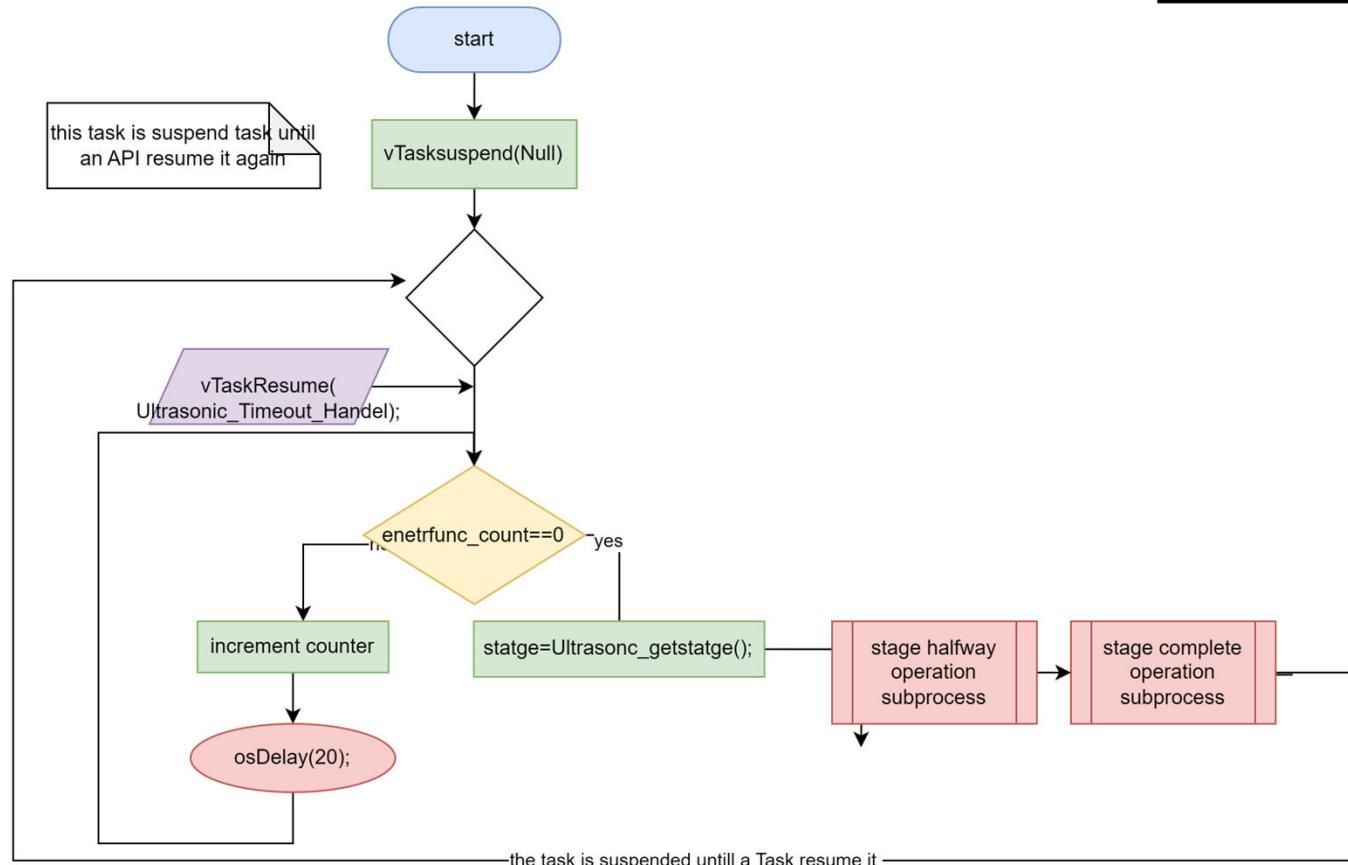
ULTRASONIC ARCHITECTURE



ULTRASONIC TASK



TIME-OUT TASK



PROBLEMS

Problems	solutions
Starvation :occurs when a task is continuously waiting for the ultrasonic sensor to measure a distance, or until a timeout period elapses. This can lead to other tasks being starved of maximum execution time up to 24 milliseconds.	Interrupt : using an additional task that comes once after (20ms) from
Voltage: inaccurate distance because ultrasonic needs 5 volt to operate correctly and the debugger supplies the kit only by 3.3 volt and the ultrasonic was connected to kit VCC	Solution: connect the Ultrasonic directly to VCC debugger to get the 5 volt required for ultrasonic

VEHICLE CONTROL

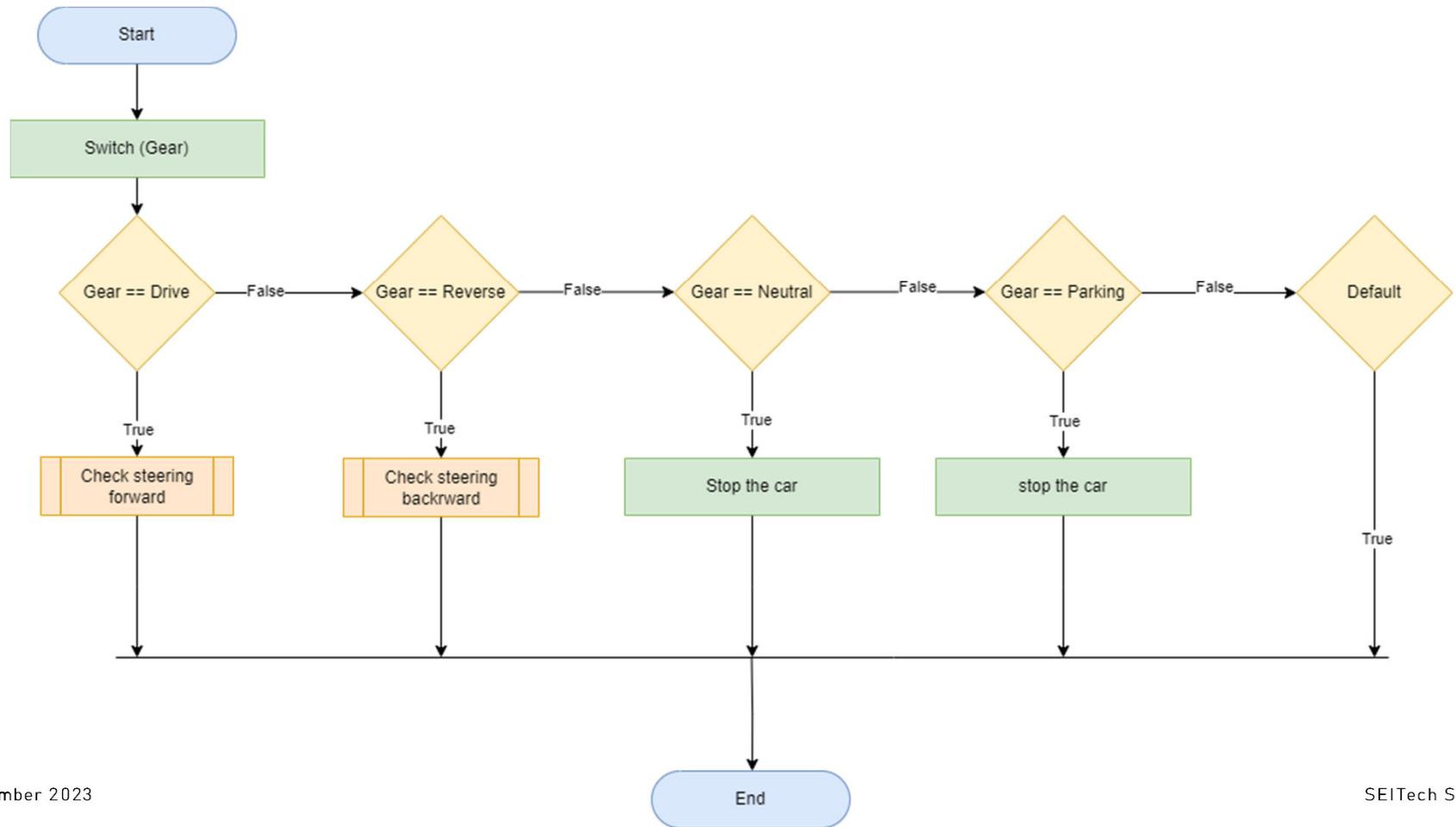
RC CAR MOVEMENT DIRECTION

The car's direction depends on 2 factors:

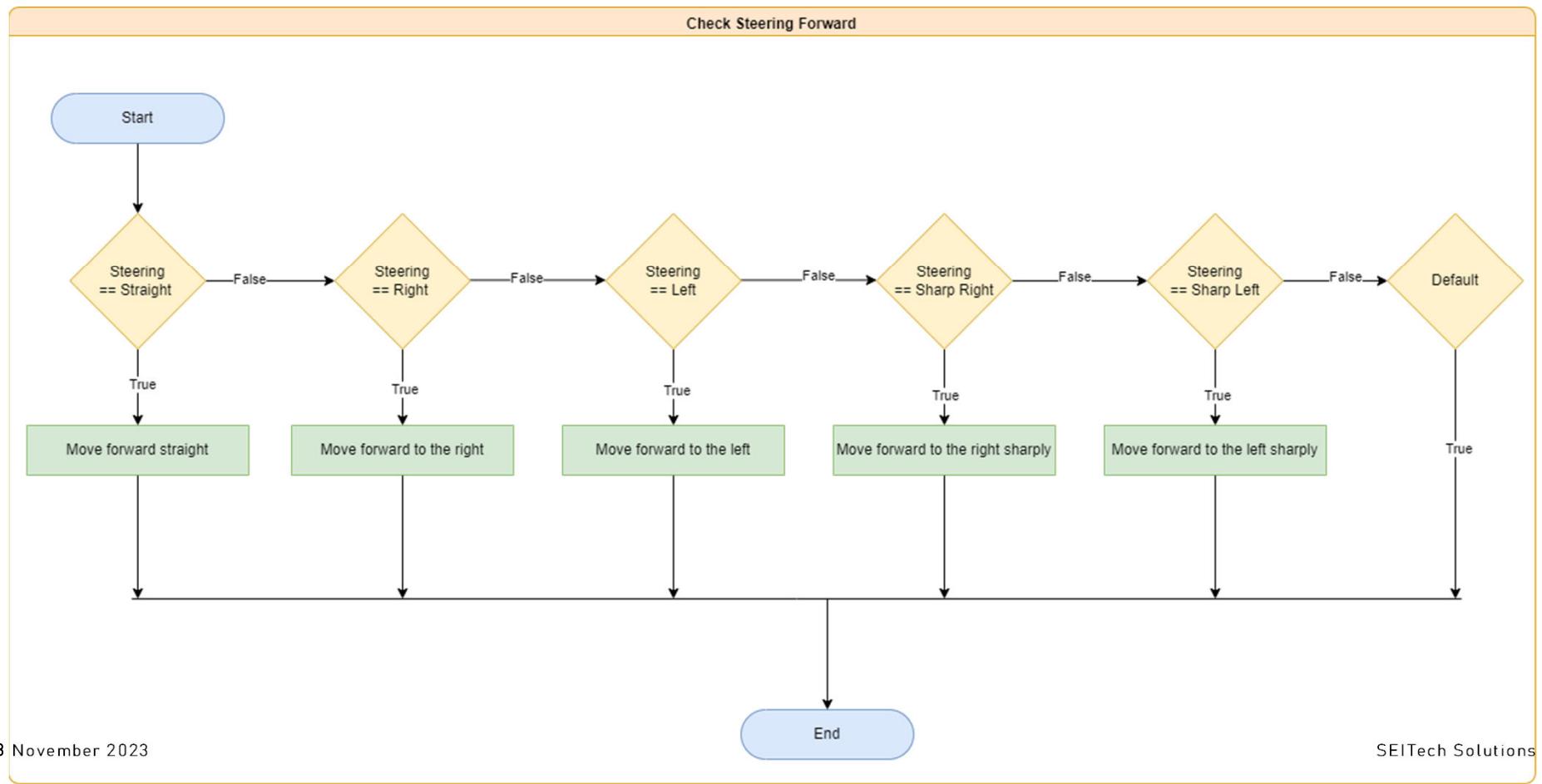
1. Steering Wheel Position
2. Gear Position



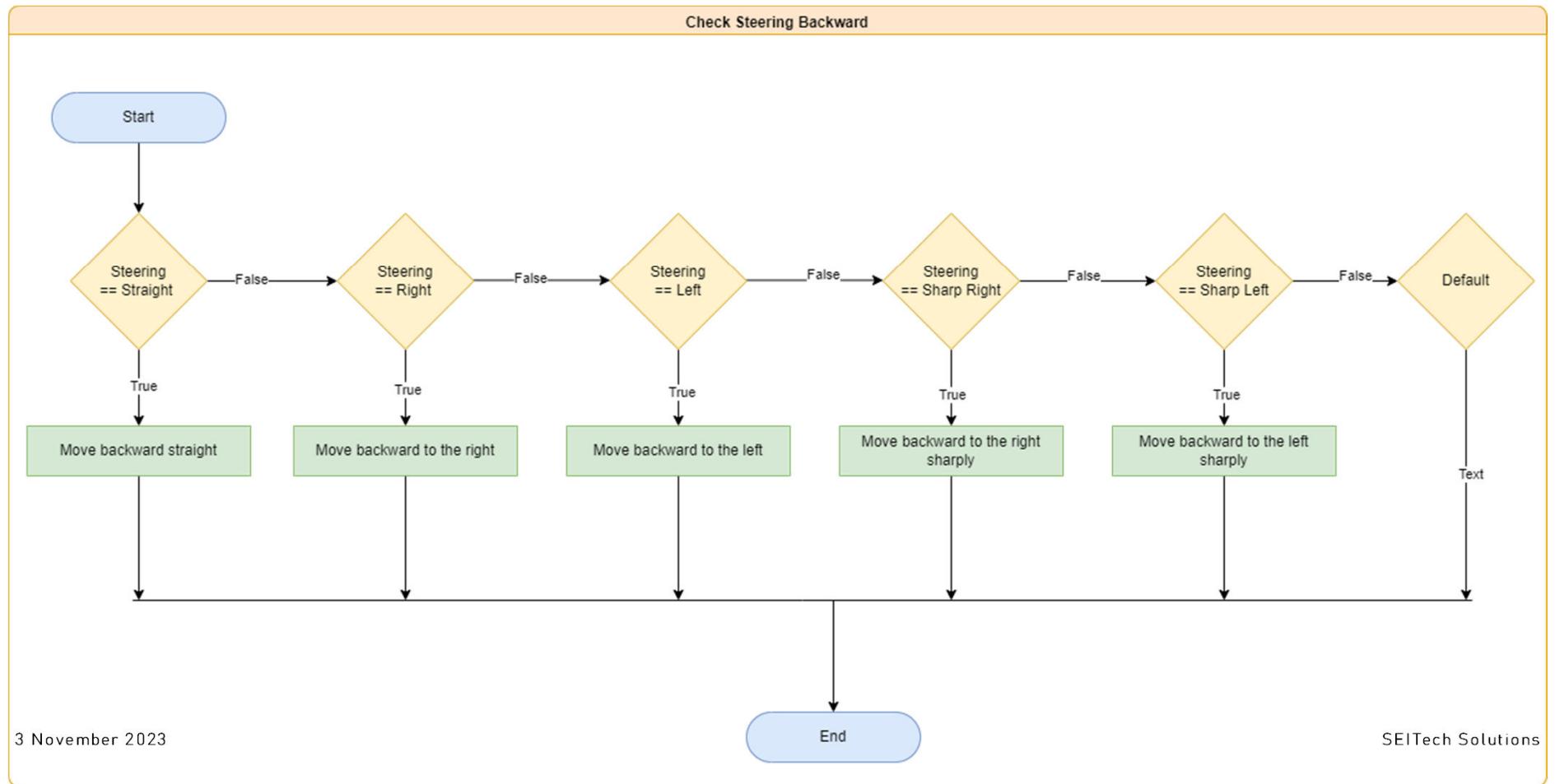
RC CAR MOVEMENT DIRECTION



RC CAR MOVEMENT DIRECTION



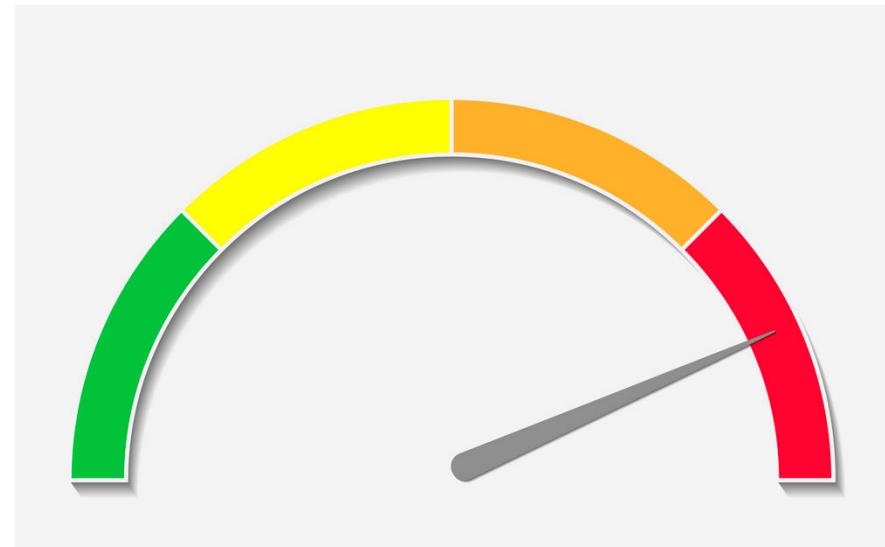
RC CAR MOVEMENT DIRECTION



RC CAR SPEED

For simplicity, we have 4 speeds:

- Speed 1
- Speed 2
- Speed 3
- Speed 4



DC MOTORS ISSUES

Random and Undefined behavior when connecting the motor's power supply.

Solution:

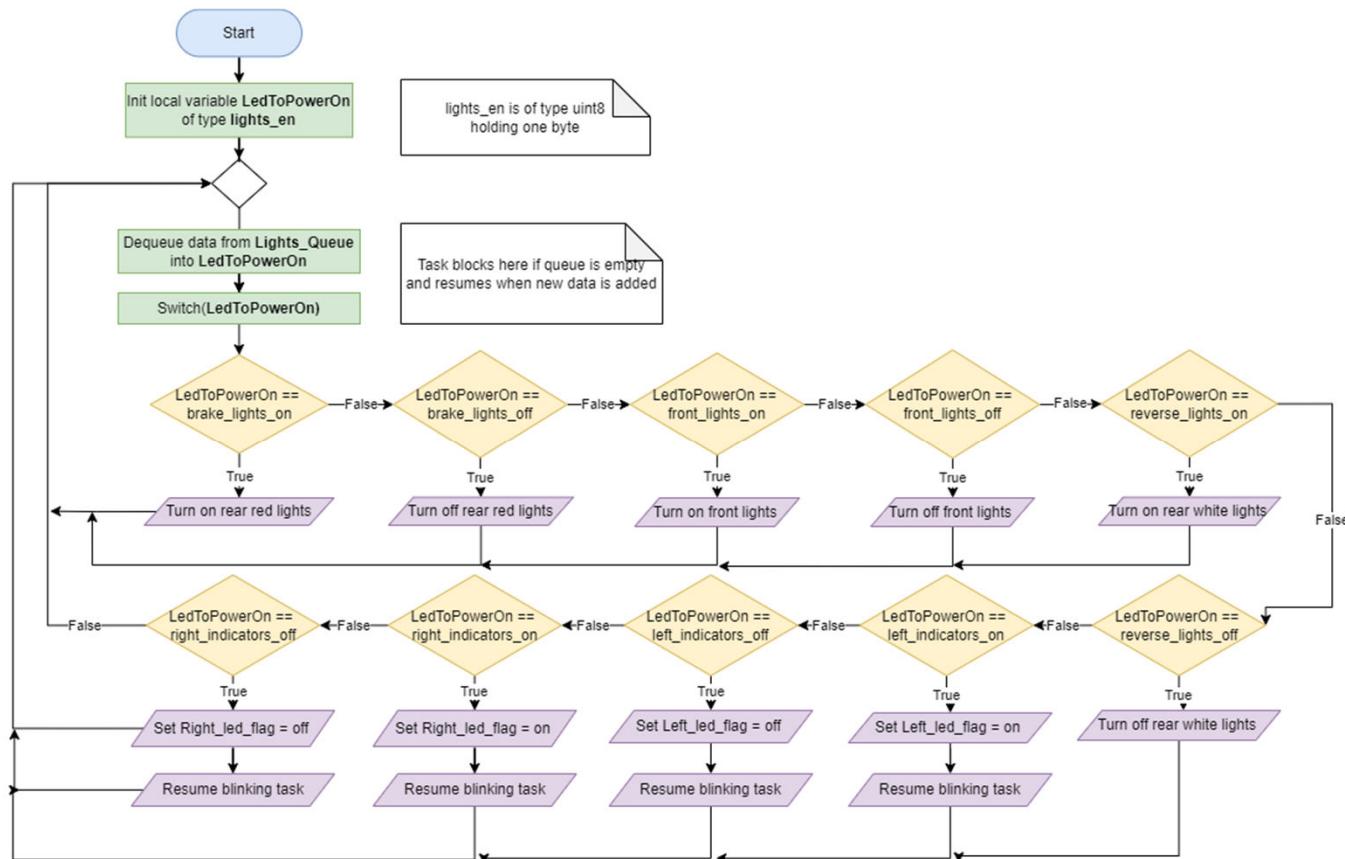


RC CAR LIGHTING SYSTEM

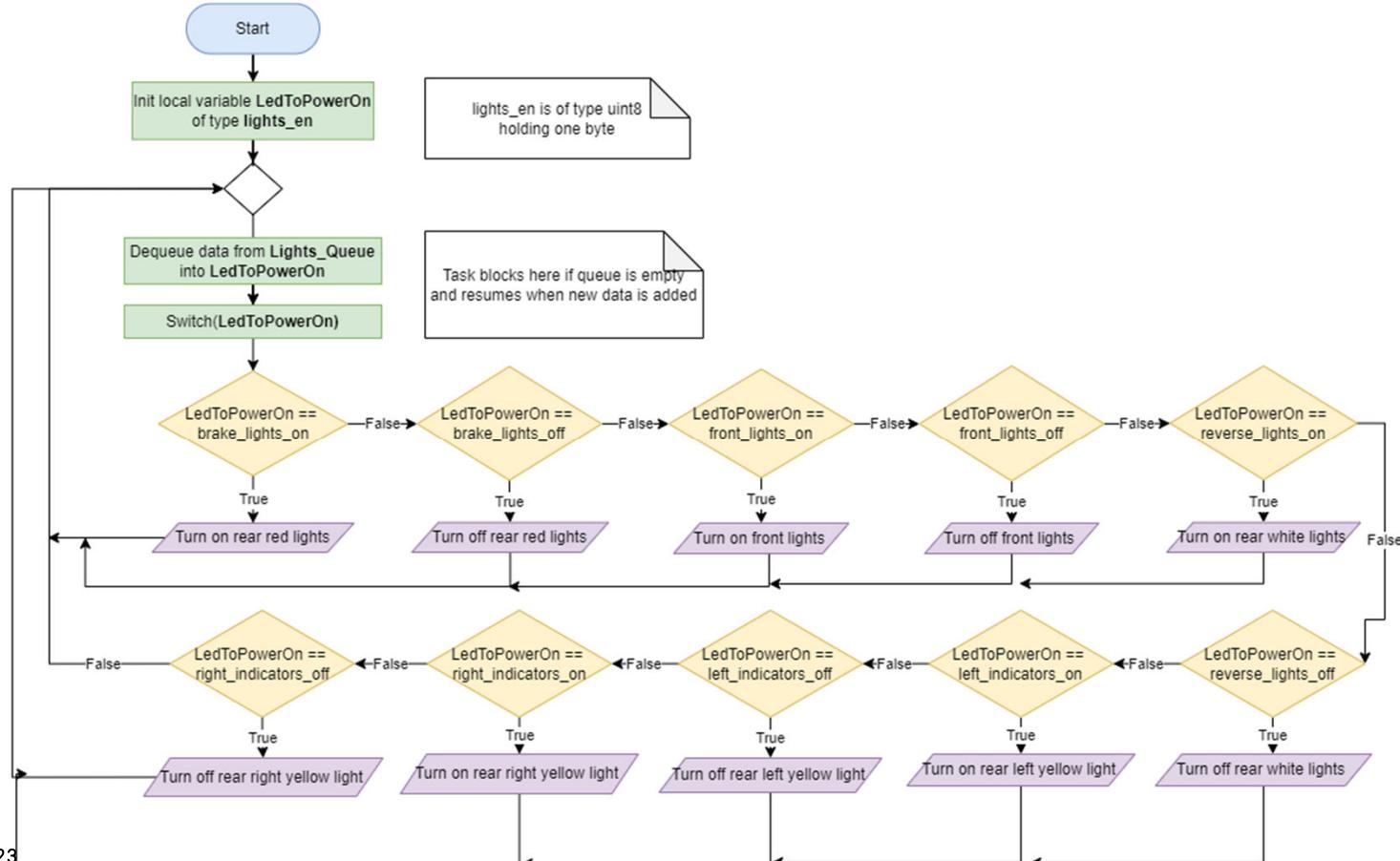
The car has 4 different lighting modes:

1. Braking lights
2. Reverse lights
3. Front lights
4. Indicator lights

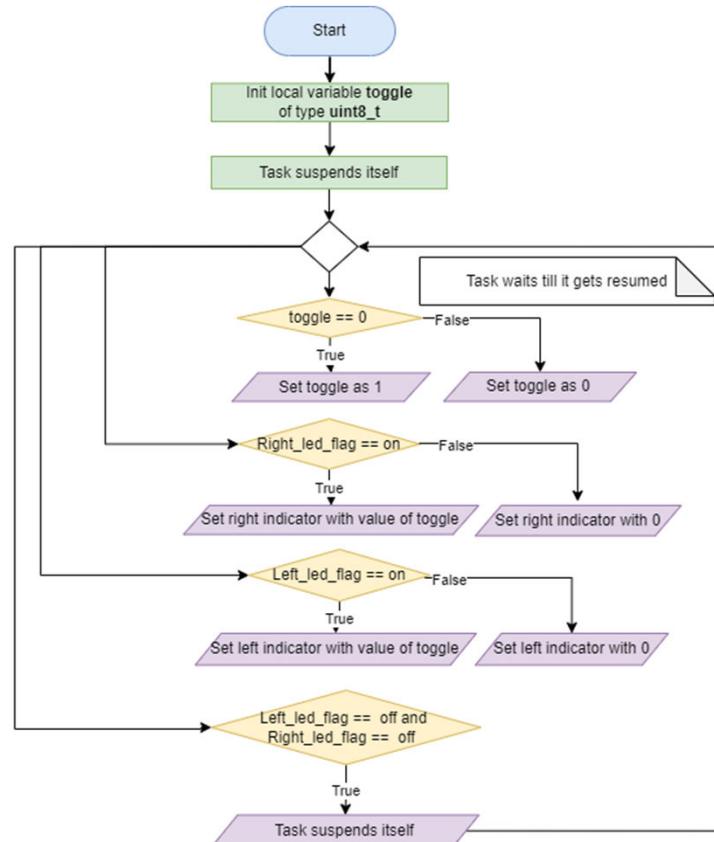
RC CAR LIGHTING TASK



RC CAR LIGHTING TASK (UPDATED)



RC CAR BLINKING TASK



FUTURE ENHANCEMENTS

- Engage brakes automatically if the car is sliding (on neutral) towards an obstacle.
- Automatic corner lights with steering.
- Replace ESP8266 with ESP32 to be able to add security features such as encrypting messages being transmitted.

THANK YOU

