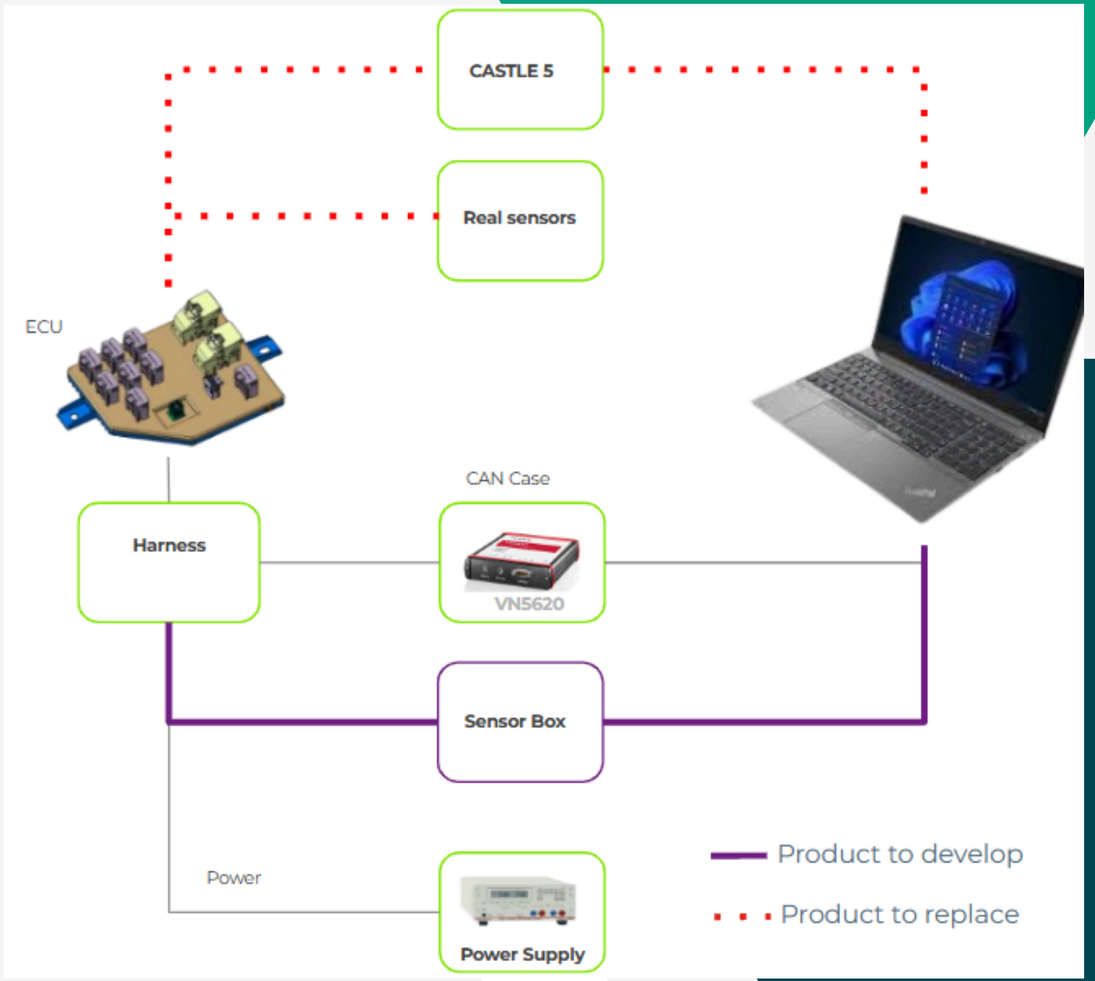
ULS Sensor Fault Simulation System

Sections

Introduction

- **Sensor Box Project:** Develop a system capable of simulating fault conditions in ultrasonic sensors
  - short to battery
  - short to ground
  - disconnected

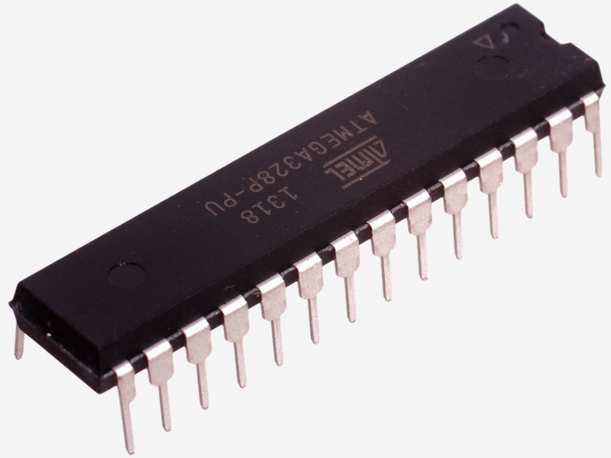


Development

Hardware Design

We used three relays to simulate the behavior of a sensor.

- The goal is to implement a signal selection system using relays, controlled by an ATmega328P microcontroller.
- This system allows selecting between different inputs and directing a single output, maintaining signal integrity and strength.
- Relays were chosen over a multiplexer because they allow direct electrical connection, ensuring better current conduction and more reliable signal delivery.



Main Components

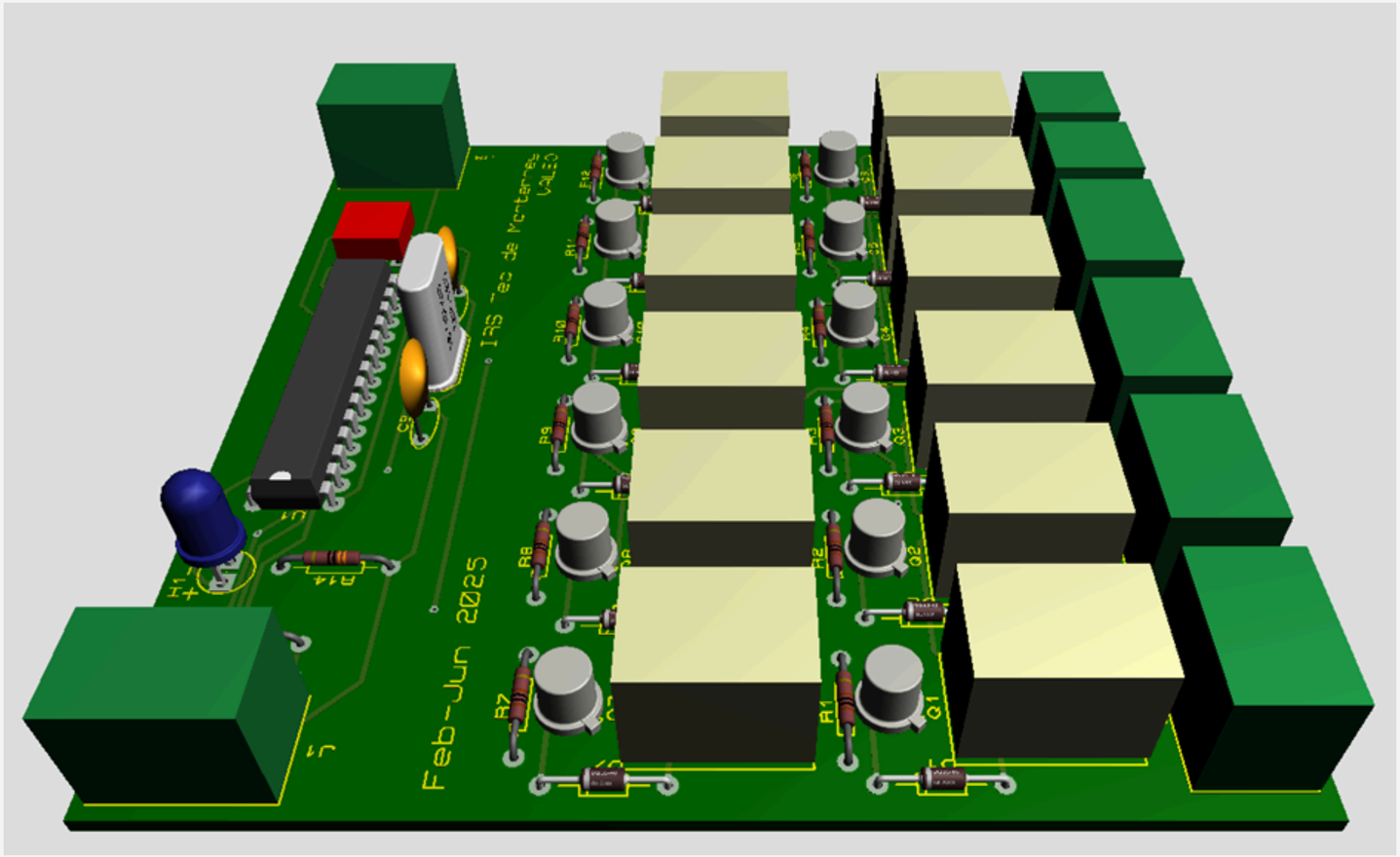
- THD-1201L Relays
- 2N2222 Transistors
- 1 kΩ Resistors
- 1N4148 Diodes
- ATmega328P Microcontroller
- 12V and 5V Power Supply

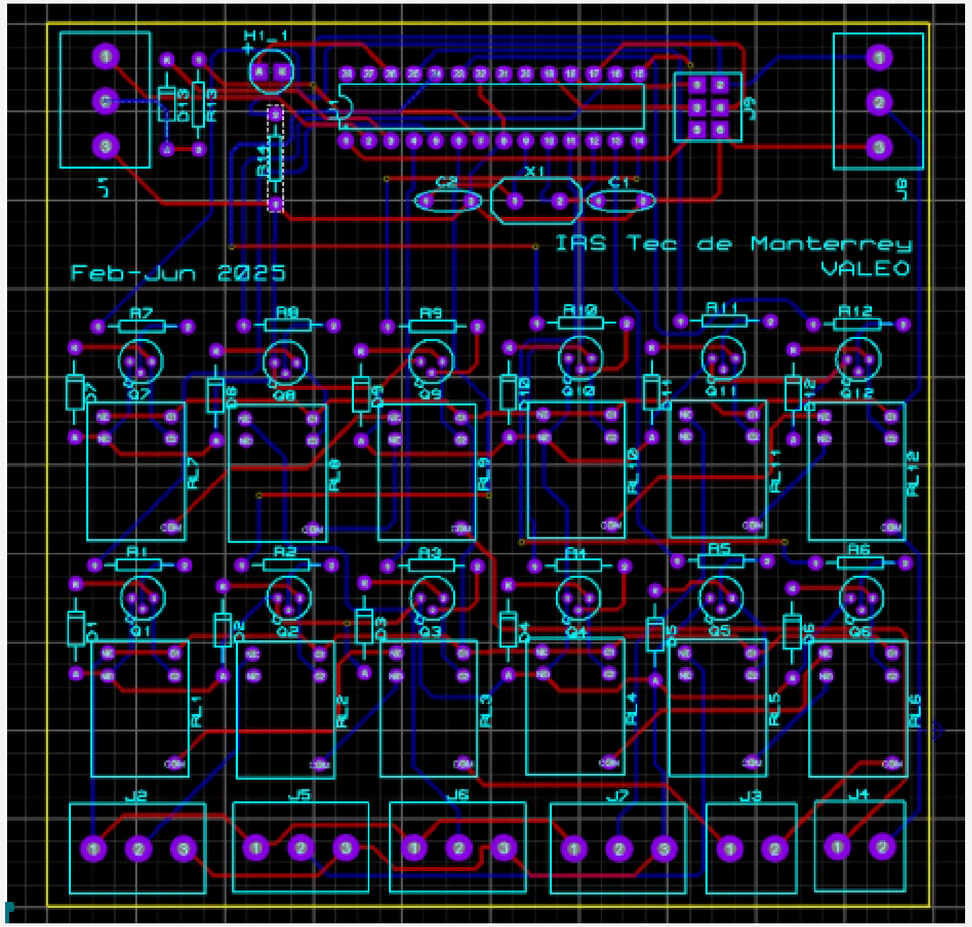
HW Setup

Assembly steps:

1. Relays were placed to simulate changing input signals.
2. 2N2222 Transistors were connected to each relay.
3. 1 kΩ Resistors were added to limit the base current of each transistor.
4. 1N4148 Diodes were placed across the coils to suppress voltage spikes.
5. The ATmega328P was programmed to control the three relays.
6. Input connections:
    - Relay 1: real sensor signal and an open line.
    - Relay 2: +12 V and GND from the power supply.
    - Outputs from these relays go to Relay 3, the final selector.
7. Power supply: 12 V for the relays and 5 V for the microcontroller.

PCB Design

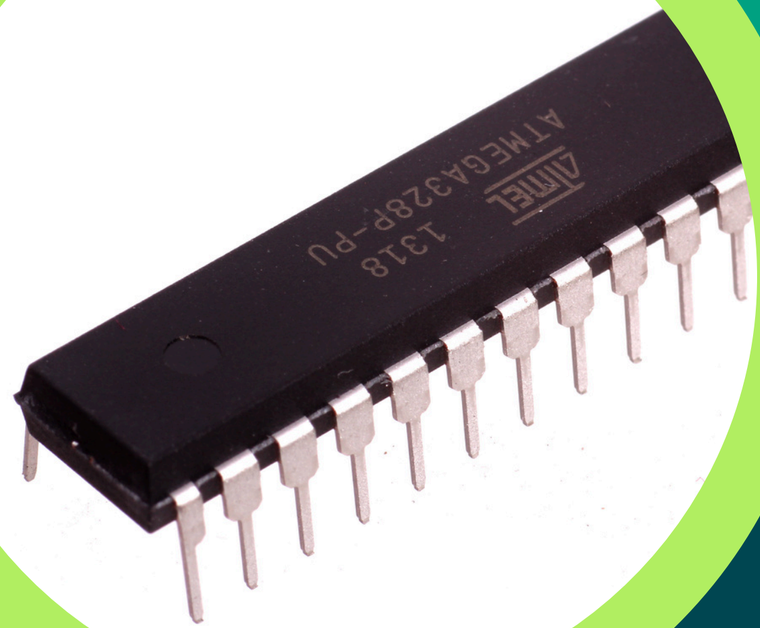




Software Development

Sensor Control via UART - ATmega328P

Serial Communication (UART)
Enables sending commands from a terminal to the ATmega328P microcontroller, which then activates the corresponding relay configuration.



Software Development

How does it work?

Each command includes:

- A sensor number
- A character indicating the desired action

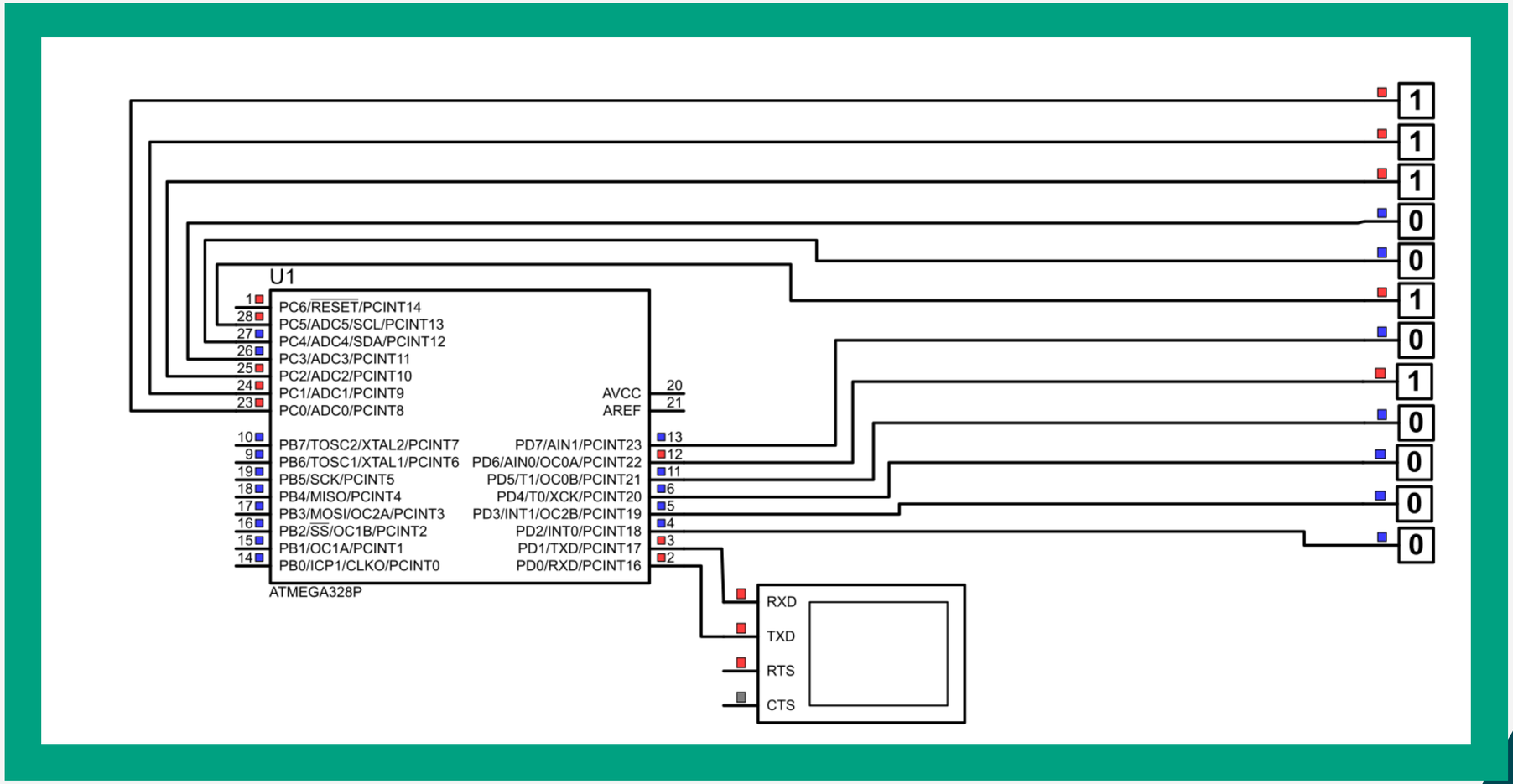
Command	Action
1v	Simulates short to battery on Sensor 1
2g	Simulates short to ground on Sensor 2
3c	Simulates an open circut
4s	Normal sensor operation

Software Development

1. Use a terminal to control the status of sensors.
2. Send commands that specify the sensor number and desired setting.
3. This enables configuration without changing the hardware.

T1	T2	T3	Status
0	0	X	+12 V
0	1	X	Gnd
1	X	0	Normal
1	X	1	Open

Software Development



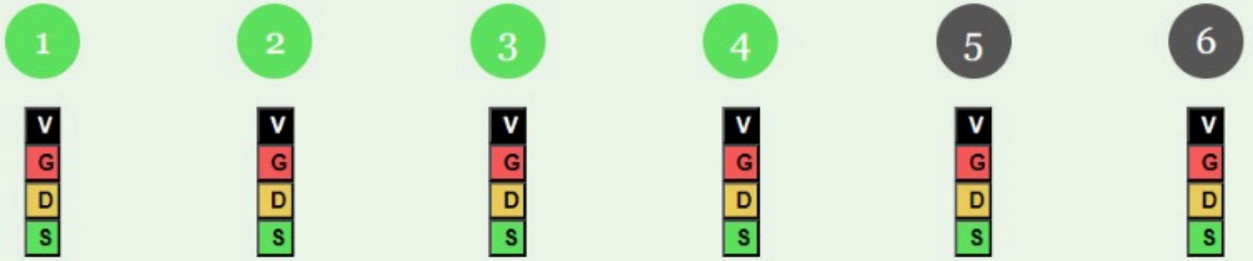
- S1: 12V
- S2: GND
- S3 Sensor
- S4: Short Circuit

Graphical User Interface

We developed a graphical interface to control the simulated sensor faults more intuitively.

The user selects the sensor and fault type using buttons.
The interface then sends the corresponding command to the microcontroller via serial communication.

Front parking sensors



Back Parking sensors

Demostration of Results

State Table

T1	T2	T3	Output
0	0	X	+12 Volts
0	1	X	GND
1	X	0	Sensor
1	X	1	Open Circuit

Conclusion

In conclusion, our project works as expected and simulates different sensor faults successfully.

We are currently demonstrating on a breadboard, but the PCB has been designed and is now being manufactured.

Additionally, the system currently works with up to 4 sensors, but a future version could be designed to support all 12 sensors, allowing full simulation for complete ECU validation.