**My Role in the Project**

In our water quality sensor project, my primary role was focused on the technical aspects, specifically testing and troubleshooting the sensors. While everyone in the team contributed to the technical work, I was responsible for conducting structured tests, debugging issues, and documenting findings. This is why I took on the Testing & Evidence section in our final report.

**Technical errors me and my team met during our project:**

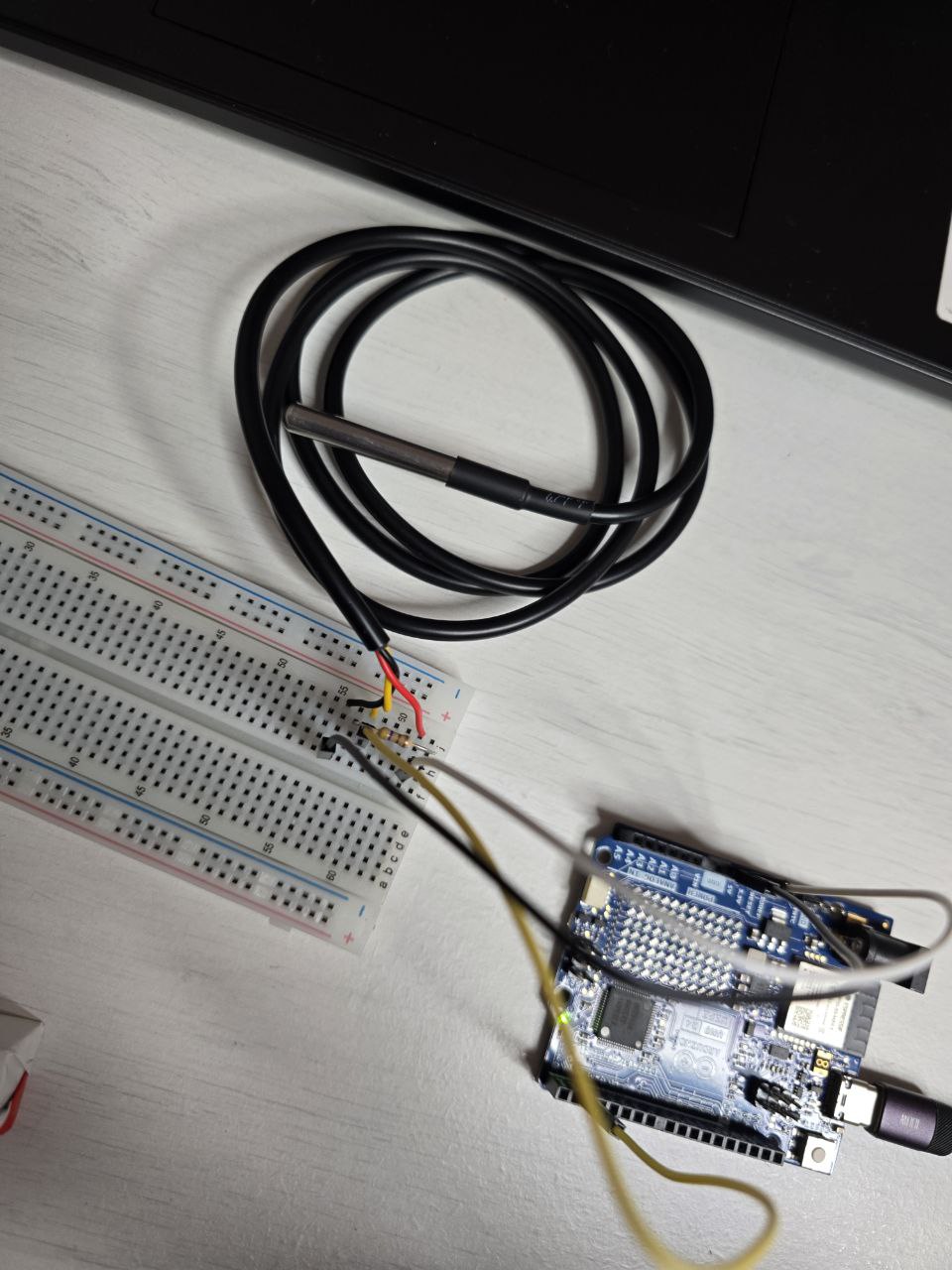
Through this project, I gained a deeper understanding of hardware compatibility, sensor calibration, and troubleshooting in embedded systems. Some key takeaways include:

* Microcontroller Compatibility: Initially, we selected the Arduino MKR WiFi 1010, but it turned out to be incompatible with our sensors due to its 3.3V limitation, forcing us to switch to the Arduino Uno R4 WiFi.
* Turbidity Sensor Issues: We struggled with the SEN0189 sensor, initially getting only 1s and 0s from the analog input. After switching to digital and correcting the formula, we obtained more reasonable readings, but they still didn’t change as expected.
* Testing Alternative Sensors: We suspected the sensor was faulty and ordered a Jopto TSW-30, but it also didn’t work under different conditions. After further research, we realized these sensors were not compatible with the Arduino Uno R4, leading us to test an Arduino Yun, which also didn’t solve the issue.
* Temperature Sensor (DS18B20) Setup: After receiving the required 4.7kΩ resistor, I attempted to set up the DS18B20 waterproof temperature sensor. Despite trying various configurations and troubleshooting methods, I kept encountering errors like "sensor not detected" or readings of -127°C. This is some photos of my attempts setting up DS18B20:

A close-up of wires on a circuit board

AI-generated content may be incorrect. Close-up of a circuit board with wires

AI-generated content may be incorrect.

A computer screen with a message

AI-generated content may be incorrect.

* Hardware Safety & Debugging: At one point, incorrect wiring caused the temperature sensor to become extremely hot, which taught me the importance of proper wiring and power management when working with Arduino.

Wires connected to a resistor

AI-generated content may be incorrect.

This is setup I used which caused the heating of sensor. After some researching, I determined it was because of putting wires too close to each other. That’s why my other set ups were different.

**Working in a Group Project**

This project reinforced the value of teamwork and collaboration in a technical environment. Some key lessons I learned about working in a group include:

* Sharing Responsibilities: Even though we all worked on the technical part, dividing specific roles helped us stay organized and efficient.
* Communication & Problem-Solving: When we faced issues (like sensor compatibility problems), discussing potential solutions as a team helped us analyze different approaches and avoid repeating mistakes.
* Time Management & Resource Sharing: Since we had only one Arduino board between five people, it was difficult to divide the work equally. To solve this, we switched every day, ensuring that each team member had the Arduino in the evening. This allowed everyone to contribute fairly.
* Trusting the Team & Staying Persistent: At times, it was frustrating not seeing immediate progress, but believing in our team’s ability to figure things out helped us stay motivated. Trusting other members to test different approaches and share their findings made the process more efficient.

**Challenges & How I Overcame Them**

* Sensor Compatibility Issues: After multiple failed attempts with different turbidity sensors, I learned the importance of checking compatibility before purchasing hardware.
* Unsuccessful Temperature Sensor Setup: I spent four hours troubleshooting the DS18B20 sensor, testing multiple setups and researching different sources. While I didn’t get it to work, I improved my debugging and critical thinking skills by systematically testing different solutions.
* Hardware Failures & Wiring Mistakes: When my incorrect wiring caused the temperature sensor to overheat, I took extra precautions moving forward, ensuring that all connections were correctly placed before powering the circuit.
* Rushing the Ordering Process: One of the biggest mistakes we made was ordering components too quickly without enough research. We should have first fully understood the setup, how everything would work, and what was needed before making purchases. Instead, we hurried and faced multiple compatibility issues. This experience taught me the importance of proper planning and thorough research before buying hardware.
* Limited Equipment: Since we only had one Arduino, we had to create a fair system to ensure everyone could work with it. By rotating access daily, we managed to keep progress steady and involve all team members equally.

**Conclusion**

Despite the technical difficulties, this project helped me develop practical skills in Arduino, sensor testing, and problem-solving. It also reinforced the importance of teamwork, adaptability, and patience when working with hardware. Most importantly, I learned that rushing into decisions without fully understanding the requirements can lead to major setbacks. Taking the time to research and plan properly is essential for a successful project. Additionally, I learned the value of trusting my team, staying persistent, and managing limited resources effectively to ensure everyone had a chance to contribute.