Day 3: Sensors and Real World Application

Dipesh Bhattarai

1 July 2025

1. Sensor Fundamentals and Types

On this day, we were introduced to the basics of sensors and how they are used to detect real-world changes like motion, distance, and light. We learned that sensors are special components that convert physical changes into electrical signals so that electronic systems like Arduino can understand and respond. We discussed some common types of sensors, including IR (infrared) sensors and ultrasonic sensors, which are often used in smart devices for tasks like detecting obstacles or measuring distance.

2. Hands-on with IR and Ultrasonic Sensors

During the hands-on session, we focused on two specific sensors: the IR sensor and the ultrasonic sensor. The **IR sensor** works by emitting infrared light and checking for reflection from a nearby object. We used it in a simple project where an **LED turns ON when an object is detected**, and turns OFF when there's nothing in front of the sensor. This helped us see in real time how sensors can be used to trigger actions. We also explored the **ultrasonic sensor**, which measures distance using sound waves. Although we didn't go deep into measurements, we understood how it's used in real-world applications like parking sensors or automated doors.

3. Analog vs Digital Sensor Data Handling

We also learned the difference between **analog** and **digital** sensor outputs. Digital sensors, like the IR sensor we used, give only two possible values — ON or OFF. Analog sensors, on the other hand, provide a range of values depending on the input they receive (like how bright or hot something is). Understanding this difference helped us know when to use

`digitalRead()` or `analogRead()` in our Arduino code. This basic concept is important when working with different types of sensors in future projects.

4. Sensor Interfacing Techniques

A key part of our learning was understanding how to **connect sensors to the Arduino** a process called interfacing. We connected the IR sensor's output pin to one of Arduino's digital input pins and also connected an LED as an output device. This helped us learn how sensors send signals and how we can use those signals to make something happen, like turning on a light. It was a simple but powerful way to see how sensors and microcontrollers communicate.

5. Recording Sensor Data

While we didn't go into detailed data recording, we learned how sensor readings can be used to control other components. In our case, we used the sensor to control an LED, which is one of the most basic ways to visually show a sensor's response. This small step helped us understand the flow of data from sensing to action and sets the foundation for more advanced tasks like logging sensor data or displaying it on a screen.

Practical Application

Overall, we learned how to **connect a sensor to an Arduino** and **write simple code to make it respond to real-world conditions**. This practical experience gave us a clear understanding of how sensors work, how they interact with hardware, and how code brings it all together. It was exciting to see something as simple as an LED respond to our movement, and it made us realize how these small systems are used in everyday technologies from automatic doors to robots. This hands-on learning built our confidence and gave us a strong starting point for more advanced electronics and automation projects.