Reflection: Challenges in Building a Mobile Robot and the Role of ROS2

Building a mobile robot involves overcoming challenges across three main domains: mechanical design, electronics, and software integration. Each area presents unique difficulties that must be addressed for the robot to function reliably.

From a **mechanical perspective**, designing a compact yet modular chassis is essential. The structure must support various components such as sensors, motors, batteries, and computing units while maintaining balance and stability. Mechanical tolerances, weight distribution, and ease of assembly are all important considerations that require precise planning.

On the **electronics side**, integrating components such as motor drivers, sensors (e.g., LiDAR, cameras), and power systems can be complex. Ensuring reliable communication between the processing unit (such as a Raspberry Pi or microcontroller) and peripherals involves careful selection of interfaces (e.g., UART, I2C, PWM) and proper power management. Debugging hardware-level issues, such as signal noise or voltage mismatches, often becomes time-consuming.

The **software** component presents perhaps the greatest challenge. Real-time control, simultaneous localization and mapping (SLAM), sensor fusion, and obstacle avoidance must all work together reliably. Managing this level of complexity without a structured framework can quickly become overwhelming.

This is where **ROS2 (Robot Operating System 2)** provides significant advantages. ROS2 enables modular software development through a publish-subscribe communication model. Each function (e.g., sensor data processing, motor control) can be developed as a separate "node" and integrated seamlessly. ROS2 also supports real-time performance and distributed computing, making it highly suitable for mobile robotics. It greatly simplifies the integration of hardware and software components, allowing developers to focus on functionality rather than low-level interfacing.