

4.2 Introduction: The Problem of Shared Resource Access

In multi-tasking systems, multiple tasks often need to access the same resource simultaneously, such as a global variable, a sensor interface, or a data structure. If this access is not managed properly, it can lead to data corruption, inconsistencies, or unexpected system behavior.

- **Race Condition**

A race condition is a situation where the outcome of a process depends on the unpredictable sequence of execution of concurrent tasks. Imagine two tasks: one is responsible for incrementing a counter variable, and the other for decrementing it. If both tasks read the counter's value, modify it, and write it back at nearly the same time, one of the updates could be lost, resulting in an incorrect final value.

To prevent race conditions, we need a mechanism that ensures only one task can access the shared resource at a time. This mechanism is known as Mutual Exclusion, or Mutex for short.

Revision #1

Created 2025-08-28 16:04:05 UTC by NS

Updated 2025-08-28 16:07:45 UTC by NS