# CWE Detail – CWE-1232

## Description

Register lock bit protection disables changes to system configuration once the bit is set. Some of the protected registers or lock bits become programmable after power state transitions (e.g., Entry and wake from low power sleep modes) causing the system configuration to be changeable.

## Extended Description

Devices may allow device configuration controls which need to be programmed after device power reset via a trusted firmware or software module (commonly set by BIOS/bootloader) and then locked from any further modification. This action is commonly implemented using a programmable lock bit, which, when set, disables writes to a protected set of registers or address regions. After a power state transition, the lock bit is set to unlocked. Some common weaknesses that can exist in such a protection scheme are that the lock gets cleared, the values of the protected registers get reset, or the lock become programmable.

## Threat-Mapped Scoring

Score: 0.0

Priority: Unclassified

## Related Attack Patterns (CAPEC)

* CAPEC-166

## Modes of Introduction

**•** Architecture and Design: N/A

**•** Implementation: N/A

## Common Consequences

**•** Impact: Modify Memory — Notes:

## Potential Mitigations

**•** Architecture and Design: Security Lock bit protections should be reviewed for behavior across supported power state transitions. Security lock programming flow and lock properties should be tested in pre-silicon and post-silicon testing including testing across power transitions. (Effectiveness: High)

## Applicable Platforms

**•** None (Class: Not Language-Specific, Prevalence: Undetermined)

## Demonstrative Examples

**•** To support the hibernate transition back to the operating state, the DRAM memory configuration must be reprogrammed even though it was locked previously. As the hibernate resume does a partial reboot, the memory configuration could be altered before the memory lock is set. Functionally the hibernate resume flow requires a bypass of the lock-based protection. The memory configuration must be securely stored and restored by trusted system firmware. Lock settings and system configuration must be restored to the same state it was in before the device entered into the hibernate mode.

**•** To resolve this issue, it is crucial to ensure that register locks are correctly initialized during the reset phase of the SoC. Correct initialization values should be established to maintain the system's integrity, security, and predictable behavior and allow for proper control of peripherals. The specifics of initializing register locks and their values depend on the SoC's design and the system's requirements; for example, access to all registers through the user privilege level should be locked at reset. To address the problem depicted in the bad code example [REF-1433], the default value for "reglk\_mem" should be set to 32'hFFFFFFFF. This ensures that access to protected data is restricted during power state transition or after reset until the system state transition is complete and security procedures have properly configured the register locks.