

전자식 차일드락 시스템 요구사항 명세, 시퀀스(UC-01~03)

1. UC-01: 전자식 잠금/해제 제어 (Basic Lock/Unlock Control)

상세 명세 (Description)

- **Actor:** 운전자(Primary), 도어 액추에이터(Secondary)
- **Pre-condition:** 시스템 전원이 ON 상태이며, ECU가 정상 동작 중임.
- **Main Flow:**
 1. 운전자가 운전석 컨트롤 패널의 차일드 락 버튼을 누름.
 2. ECU가 현재 시스템의 잠금 상태(State)를 확인 함.
 3. 현재 '해제' 상태라면 '잠금' 명령을, '잠금' 상태라면 '해제' 명령을 결정함.
 4. ECU가 뒷좌석 도어 액추에이터로 구동 신호를 전송함.
 5. 액추에이터가 내부 핸들의 물리적 링크를 연결 또는 차단함.
- **Post-condition:** 도어 상태가 변경되고 클러스터에 현재 상태가 표시됨.

시퀀스 설계 (Sequence Flow)

1. **Driver** → **Button**: 버튼 누름 (Click Event)
 2. **Button** → **ECU**: 신호 전달 (Signal: Toggle)
 3. **ECU** → **ECU**: 상태 판별 (Condition: Is Speed < 3km/h? 등 간단한 체크)
 4. **ECU** → **Actuator**: 구동 명령 (Command: Lock/Unlock)
 5. **Actuator** → **ECU**: 완료 피드백 (Ack: Done)
 6. **ECU** → **Cluster**: 상태 업데이트 (Display: Lock/Unlock Status)
-

2. UC-02: 차속 기반 해제 방지 로직 (Speed-Dependent Unlock Prevention)

상세 명세 (Description)

- **Actor:** 운전자(Primary), 차속 센서(Secondary)
- **Pre-condition:** 시스템이 '잠금(Locked)' 상태이며, 차량이 주행 중임.
- **Main Flow:**
 1. 주행 중 운전자가 실수 또는 의도적으로 해제 버튼을 누름.
 2. ECU가 차속 센서로부터 실시간 차량 속도 데이터를 수신함.
 3. ECU가 수신된 속도를 임계값(예: 3km/h)과 비교함.
 4. 속도가 3km/h를 초과하므로 ECU는 해제 명령을 거부(Ignore)하기로 결정함.
 5. ECU는 액추에이터를 구동하지 않고 유지함.
- **Post-condition:** 잠금 상태가 유지되며, 클러스터에 "주행 중 해제 불가" 경고를 송출함.

시퀀스 설계 (Sequence Flow)

1. **Vehicle Sensor** → **ECU:** 차속 데이터 전송 (Data: Speed > 3km/h)
2. **Driver** → **Button:** 해제 버튼 누름 (Request: Unlock)
3. **Button** → **ECU:** 신호 전달
4. **ECU** → **ECU:** 로직 판단 (Logic: If Speed > Threshold, Deny Unlock)
5. **ECU** → **Cluster:** 경고 메시지 송출 (Display: Warning - Vehicle Moving)
6. **ECU** (No Signal) **Actuator:** 액추에이터 구동 안 함

3. UC-03: 안전 하차 보조 연동 (SEA - Safe Exit Assist)

상세 명세 (Description)

- **Actor:** 운전자(Primary), 후측방 레이더 센서(Secondary), 클러스터(Secondary)
- **Pre-condition:** 차량이 정차(0km/h)했으나, 후측방에서 타 차량이나 자전거가 접근 중임.
- **Main Flow:**
 1. 운전자가 하차를 위해 해제 버튼을 누름.
 2. ECU가 후측방 레이더(BSD) 센서로부터 위험 객체 감지 신호를 수신함.
 3. ECU는 충돌 위험이 있다고 판단하여 해제 명령을 강제로 차단함.
 4. ECU는 잠금 상태를 유지(Active Lock)함.

5. ECU는 즉시 경고음을 발생시키고 클러스터에 위험 알림을 표시함.
- **Post-condition:** 위험 요소가 사라질 때까지 잠금이 해제되지 않아 개문 사고를 방지함.

시퀀스 설계 (Sequence Flow)

1. **Radar Sensor** → **ECU**: 위험 객체 접근 감지 (Data: Object Detected)
 2. **Driver** → **Button**: 해제 버튼 누름 (Request: Unlock)
 3. **Button** → **ECU**: 신호 전달
 4. **ECU** → **ECU**: 안전 판단 (Logic: Hazard Present = True)
 5. **ECU** → **Cluster**: 긴급 경고 메시지 및 경고음 (Display & Sound: Hazard Warning)
 6. **ECU** (Force Hold) **Actuator**: 잠금 상태 강제 유지
-

UML 작성 가이드 (Draw.io 활용 시)

- **Sequence Diagram**을 그릴 때 상단의 객체(Lifeline) 순서는 다음과 같이 배치하는 것이 표준적입니다:
 - **Driver** | **Button** | **ECU** | **Sensors(Speed/Radar)** | **Actuator** | **Cluster**
- **Control Logic:** **Alt** (Alternative) 또는 **Opt** (Optional) 박스를 사용하여 속도 조건이나 위험 감지 유무에 따른 분기 로직을 표현하세요.