Exterior Lighting — Customer Function Requirements

(SLH-style demonstration document for Signal Orbit parsing)

Document ID: SLH-DEMO-EL-REQ **Version:** v1.0 **Date:** 25 Nov 2025

Classification: Demonstration / Synthetic dataset (non-confidential)

This document intentionally reads like an OEM-style requirements specification (similar in tone and structure to SLH documents). It is generated from the Signal Orbit *master dataset* (systems, ECUs, buses, signals, signal paths, owners, ownerships) and presents the same information as a narrative requirements story rather than a direct data dump.

Revision History

Version	Date	Author	Notes
v1.0	25 Nov 2025	Signal Orbit (generated)	Initial SLH-style narrative from Excel master dataset

1 Purpose and Scope

The purpose of this document is to describe a set of exterior-lighting customer functions using clear, tool-friendly language. The document can be consumed by people (as a readable specification) and by tools (as a structured source that can be parsed into a graph).

The scope includes: (a) behaviours such as indicators, main light, high beam, fog lamps, DRL and welcome/goodbye; (b) the participating systems/ECUs; (c) the network interfaces (CAN/LIN/Ethernet) and representative signals; (d) example signal paths and ownership responsibilities.

Out of scope: country-specific legal constraints, homologation details, lamp photometry, and production-level timing calibration. All timing values in this document are illustrative and intended for demonstrations.

2 Systems Overview

Exterior lighting is implemented as a collaboration between body-domain controllers, lamp driver units, and HMI components. The following subsections describe the logical systems and how requests flow through the architecture.

2.1 High-level narrative

Driver requests (stalk actions, hazard switch, light rotary switch) are interpreted in the body domain and forwarded into the lighting control chain. Vehicle-state information (speed, brake pressure, gear selection), sensor information (ambient light), and power limitations further constrain what is allowed to operate. A lighting master coordinates patterns and delegates actuation to dedicated lamp control units. Finally, the instrument cluster and infotainment receive feedback to display tell■tales and HMI status.

In this demonstration dataset, the primary coordinating nodes are the **Exterior Light Control ECU (ELC)** and the **Light System Master (LSM)**. Lamp actuation is performed by a headlamp controller (**HCU**) and a rear-lamp controller (**RLCU**) over LIN sub-buses. The **Instrument Cluster (IC)** and **Infotainment Head Unit (IHU)** provide driver feedback.

2.2 Network overview

- `CAN_BODY` — CAN Body (CAN, 500 kbps)
- `CAN_CHASSIS` — CAN Chassis (CAN, 500 kbps)
- `LIN_FRONT` — LIN Front Lamps (LIN, 20 kbps)
- `LIN_REAR` — LIN Rear Lamps (LIN, 20 kbps)
- `ETH_BACKBONE` — Ethernet Backbone (ETHERNET, 100000 kbps)

2.3 Participating systems (logical)

- `ADAS` — **ADAS Domain Controller**. Sensor fusion & driver assistance
- `RPE` — **Remote Parking ECU**. Remote/automatic parking maneuvers
- `SVS` — **Surround View System**. Camera system for front/rear/360° view
- `BCM` — **Body Control Module**. Body electronics, stalks, hazard switch, power supply
- `DLC_FL` — **Door Light Controller FL**. Front-left door/mirror/puddle lights
- `DLC_FR` — **Door Light Controller FR**. Front-right door/mirror/puddle lights

- ESP_BRS — **Brake / ESP ECU**. Brake system, ESP, deceleration sensing
- TELE — **Telematics ECU**. Connectivity, remote commands, logging
- IC — **Instrument Cluster**. Instrument cluster, tell-tales
- IHU — **Infotainment Head Unit**. Infotainment, personalization, vehicle settings
- GATE — **Gateway ECU**. Vehicle gateway between buses
- AFS — **Adaptive Front Lighting ECU**. Adaptive front lighting actuators
- ELC — **Exterior Light Control**. Central exterior lighting logic (functions, arbitration)
- HCU — **Headlamp Control Unit**. Front headlamp and DRL driver
- ILC — **Interior Light Controller**. Interior / ambient lighting controller
- LSM — **Light System Master**. Lighting system master / light switch logic
- RLCU — **Rear Light Control Unit**. Rear lamp driver (tail, stop, indicator, fog, reverse)
- PMU — **Power Management Unit**. 12V power, load shedding, derating

2.4 Ownership model

Each logical system and each customer function has an assigned owner team. Ownership is used for routing clarifications, defect triage, and change approvals. Where multiple teams contribute, the primary owner is responsible for end-to-end acceptance of the function behaviour.

- **Exterior Light Control Team** (OWN_ELC) — elc-team@example.com
- **Body Control Module Team** (OWN_BCM) — bcm-team@example.com
- **Light System Master Team** (OWN_LSM) — lsm-team@example.com
- **Headlamp Control Unit Team** (OWN_HCU) — hcu-team@example.com
- **Rear Light Control Unit Team** (OWN_RLCU) — rlcu-team@example.com
- **Adaptive Front Lighting ECU Team** (OWN_AFS) — afs-team@example.com
- **ADAS Domain Controller Team** (OWN_ADAS) — adas-team@example.com
- **Brake / ESP ECU Team** (OWN_ESP_BRS) — esp_brs-team@example.com
- **Power Management Unit Team** (OWN_PMU) — pmu-team@example.com
- **Surround View System Team** (OWN_SVS) — svs-team@example.com
- **Instrument Cluster Team** (OWN_IC) — ic-team@example.com
- **Infotainment Head Unit Team** (OWN_IHU) — ihu-team@example.com
- **Telematics ECU Team** (OWN_TELE) — tele-team@example.com
- **Gateway ECU Team** (OWN_GATE) — gate-team@example.com
- **Door Light Controller FL Team** (OWN_DLC_FL) — dlc_fl-team@example.com
- **Door Light Controller FR Team** (OWN_DLC_FR) — dlc_fr-team@example.com
- **Interior Light Controller Team** (OWN_ILC) — ilc-team@example.com
- **Remote Parking ECU Team** (OWN_RPE) — rpe-team@example.com

3 Customer Function Specifications

Each customer function below is presented in an SLH-style structure with: overview, operating conditions, behaviour, and interfaces. Requirements are phrased using 'shall' statements. Identifiers are stable and can be used for traceability in Signal Orbit.

3.1 Left indicator / turn signal (CF-BLINKER_LEFT-v1)

Left indicator / turn signal describes the customer-visible behaviour for `BLINKER_LEFT`. Control of left-side front and rear indicators based on stalk or hazard requests.

3.1.1 Identifier and overview

Function ID: `CF-BLINKER_LEFT-v1`
Function code: `BLINKER_LEFT`
Domain: lighting
Logical owner: **Exterior Light Control Team** (elc-team@example.com)

3.1.2 Scenario narrative

This function can be understood as a message journey through the vehicle: a request is detected, validated against enabling conditions, translated into lamp commands, and finally reported back to the driver via tell■tales and HMI status.

Involved ECUs (in typical order):

- Body Control Module ECU (`BCM`)
- Exterior Light Control ECU (`ELC`)
- Light System Master ECU (`LSM`)
- Headlamp Control Unit ECU (`HCU`)
- Rear Light Control Unit ECU (`RLCU`)
- Instrument Cluster ECU (`IC`)
- Infotainment Head Unit ECU (`IHU`)

3.1.3 Operating conditions

- Vehicle electrical system shall be in a driving-capable or awake state (ignition/ready or permitted wake-up).
- Signals required for decision-making shall be valid (no stale timestamps, no implausible values).
- If a global lighting power limitation is active (`POWER_LIMIT`), optional functions shall be inhibited or degraded according to configuration.
- A driver request shall be present via stalk/switch input or configured automatic mode.

3.1.4 Functional behaviour

- When the driver selects the left indicator, the body controller reports a left-turn request to the lighting controller. The lighting chain then generates a periodic blink pattern and distributes it to both front and rear left indicators.
- The instrument cluster shall display the left indicator tell■tale synchronized with the lamp pattern.

- The infotainment HMI may display a consolidated lighting status for diagnostics and user feedback.

3.1.5 Normative requirements

- **REQ-BLINKER_LEFT-001** — The function shall evaluate all operating conditions before activating any lamp output.
- **REQ-BLINKER_LEFT-002** — The function shall accept driver/input requests via the representative input signals listed in the interfaces section.
- **REQ-BLINKER_LEFT-004** — The function shall translate the decision into lamp command outputs to the applicable lamp driver ECUs.
- **REQ-BLINKER_LEFT-005** — The function shall report status to HMI, including tell■tales where applicable.

3.1.6 Interfaces and signal exchange

Representative request / driver input signals:

- `STALK_TURN_LEFT` — Left turn stalk request (BCM→ELC)
- `HAZARD_SWITCH` — Hazard switch status (BCM→LSM)

Representative lamp command signals:

- `CMD_BLINKER_LEFT_FRONT` — Command left front indicator (ELC→HCU)
- `CMD_BLINKER_LEFT_REAR` — Command left rear indicator (ELC→RLCU)
- `CMD_HAZARD_ALL` — Command hazard pattern all lamps (LSM→RLCU)

Representative HMI / status signals:

- `TT_BLINKER_LEFT` — Tell-tale left indicator (ELC→IC)
- `TT_HAZARD` — Tell-tale hazard (LSM→IC)
- `HMI_LIGHT_STATUS` — Lighting status to infotainment (ELC→IHU)

Representative signal path (hop-by-hop):

- 1. `BCM` → `ELC` via `CAN_BODY` (CAN, 10 ms)
- 2. `ELC` → `LSM` via `CAN_BODY` (CAN, 10 ms)
- 3. `LSM` → `HCU` via `LIN_FRONT` (LIN, 15 ms)
- 4. `LSM` → `RLCU` via `LIN_REAR` (LIN, 15 ms)
- 5. `ELC` → `IC` via `CAN_BODY` (CAN, 20 ms)
- 6. `ELC` → `IHU` via `ETH_BACKBONE` (ETHERNET, 30 ms)

3.1.7 Diagnostics and fault handling

If required inputs are invalid, the function shall fail safe (no unintended lamp activation) and expose a diagnostic state via HMI/status reporting. If a commanded lamp driver ECU does not acknowledge a request (bus off, LIN timeout), the function shall continue operating with remaining available outputs and log a fault condition.

3.1.8 Acceptance criteria (example)

- Given valid enabling conditions, the commanded lamps reach the expected on/off pattern within the indicative latency budget.

- Given a removed request or disabled condition, lamps return to OFF within the configured exit timing.
- Given an active function, HMI state matches lamp output state (no contradiction between tell■tale and lamps).

3.2 Brake light (CF-BRAKE_LIGHT-v1)

Brake light describes the customer-visible behaviour for `BRAKE_LIGHT`. Activation of rear brake lamps when brake system reports deceleration or pedal request.

3.2.1 Identifier and overview

Function ID: `CF-BRAKE_LIGHT-v1`
Function code: `BRAKE_LIGHT`
Domain: lighting
Logical owner: **Exterior Light Control Team** (elc-team@example.com)

3.2.2 Scenario narrative

This function can be understood as a message journey through the vehicle: a request is detected, validated against enabling conditions, translated into lamp commands, and finally reported back to the driver via tell■tales and HMI status.

Involved ECUs (in typical order):

- Brake / ESP ECU ECU (`ESP_BRS`)
- Exterior Light Control ECU (`ELC`)
- Rear Light Control Unit ECU (`RLCU`)
- Instrument Cluster ECU (`IC`)

3.2.3 Operating conditions

- Vehicle electrical system shall be in a driving-capable or awake state (ignition/ready or permitted wake-up).
- Signals required for decision-making shall be valid (no stale timestamps, no implausible values).
- If a global lighting power limitation is active (`POWER_LIMIT`), optional functions shall be inhibited or degraded according to configuration.
- Brake actuation shall be detected via brake/ESP domain (e.g., brake pressure or deceleration signal).

3.2.4 Functional behaviour

- When the vehicle detects a braking event, brake lamps shall be illuminated to warn following traffic. Brake light activation shall follow the vehicle-state domain signals and be independent of driver stalk inputs.
- Brake lamps shall remain active while braking is present and shall switch off when braking is no longer present.

3.2.5 Normative requirements

- **REQ-BRAKE_LIGHT-001** — The function shall evaluate all operating conditions before activating any lamp output.

- **REQ-BRAKE_LIGHT-003** — The function shall use vehicle-state/sensor inputs (e.g., speed, brake pressure, ambient light) to gate or adapt behaviour.
- **REQ-BRAKE_LIGHT-004** — The function shall translate the decision into lamp command outputs to the applicable lamp driver ECUs.
- **REQ-BRAKE_LIGHT-005** — The function shall report status to HMI, including tell■tales where applicable.

3.2.6 Interfaces and signal exchange

Representative state / sensor / power signals:

- `BRAKE_PRESSURE` — Brake pressure / decel (ESP_BRS→ELC)

Representative lamp command signals:

- `CMD_BRAKE_LIGHT` — Command brake lights (ELC→RLCU)

Representative HMI / status signals:

- `HMI_LIGHT_STATUS` — Lighting status to infotainment (ELC→IHU)

Representative signal path (hop-by-hop):

- 1. `ESP_BRS` → `ELC` via `CAN_CHASSIS` (CAN, 10 ms)
- 2. `ELC` → `RLCU` via `LIN_REAR` (LIN, 15 ms)
- 3. `ELC` → `IC` via `CAN_BODY` (CAN, 20 ms)

3.2.7 Diagnostics and fault handling

If required inputs are invalid, the function shall fail safe (no unintended lamp activation) and expose a diagnostic state via HMI/status reporting. If a commanded lamp driver ECU does not acknowledge a request (bus off, LIN timeout), the function shall continue operating with remaining available outputs and log a fault condition.

3.2.8 Acceptance criteria (example)

- Given valid enabling conditions, the commanded lamps reach the expected on/off pattern within the indicative latency budget.
- Given a removed request or disabled condition, lamps return to OFF within the configured exit timing.
- Given an active function, HMI state matches lamp output state (no contradiction between tell■tale and lamps).

3.3 Rear fog light (CF-REAR_FOG-v1)

Rear fog light describes the customer-visible behaviour for `REAR_FOG`. Control of rear fog light for reduced visibility conditions.

3.3.1 Identifier and overview

Function ID: `CF-REAR_FOG-v1`
Function code: `REAR_FOG`
Domain: lighting
Logical owner: **Exterior Light Control Team** (elc-team@example.com)

3.3.2 Scenario narrative

This function can be understood as a message journey through the vehicle: a request is detected, validated against enabling conditions, translated into lamp commands, and finally reported back to the driver via tell■tales and HMI status.

Involved ECUs (in typical order):

- Body Control Module ECU (BCM)
- Exterior Light Control ECU (ELC)
- Rear Light Control Unit ECU (RLCU)
- Instrument Cluster ECU (IC)

3.3.3 Operating conditions

- Vehicle electrical system shall be in a driving-capable or awake state (ignition/ready or permitted wake-up).
- Signals required for decision-making shall be valid (no stale timestamps, no implausible values).
- If a global lighting power limitation is active (POWER_LIMIT), optional functions shall be inhibited or degraded according to configuration.
- A driver request shall be present via stalk/switch input or configured automatic mode.

3.3.4 Functional behaviour

- Rear fog lights provide enhanced rear visibility in adverse weather. When enabled by the driver, the rear fog lamp shall illuminate subject to configuration rules and base lighting prerequisites.
- The instrument cluster shall indicate rear-fog status when the rear fog function is active.

3.3.5 Normative requirements

- **REQ-REAR_FOG-001** — The function shall evaluate all operating conditions before activating any lamp output.
- **REQ-REAR_FOG-002** — The function shall accept driver/input requests via the representative input signals listed in the interfaces section.
- **REQ-REAR_FOG-004** — The function shall translate the decision into lamp command outputs to the applicable lamp driver ECUs.
- **REQ-REAR_FOG-005** — The function shall report status to HMI, including tell■tales where applicable.

3.3.6 Interfaces and signal exchange

Representative request / driver input signals:

- STALK_FOG_REAR — Rear fog request (BCM→ELC)

Representative lamp command signals:

- CMD_REAR_FOG — Command rear fog lamp (ELC→RLCU)

Representative HMI / status signals:

- TT_FOG_REAR — Tell-tale rear fog (ELC→IC)
- HMI_LIGHT_STATUS — Lighting status to infotainment (ELC→IHU)

Representative signal path (hop-by-hop):

- 1. BCM → ELC via CAN_BODY (CAN, 10 ms)

- 2. ELC → RLCU via LIN_REAR (LIN, 15 ms)
- 3. ELC → IC via CAN_BODY (CAN, 20 ms)

3.3.7 Diagnostics and fault handling

If required inputs are invalid, the function shall fail safe (no unintended lamp activation) and expose a diagnostic state via HMI/status reporting. If a commanded lamp driver ECU does not acknowledge a request (bus off, LIN timeout), the function shall continue operating with remaining available outputs and log a fault condition.

3.3.8 Acceptance criteria (example)

- Given valid enabling conditions, the commanded lamps reach the expected on/off pattern within the indicative latency budget.
- Given a removed request or disabled condition, lamps return to OFF within the configured exit timing.
- Given an active function, HMI state matches lamp output state (no contradiction between tell■tale and lamps).

3.4 High beam (CF-HIGH_BEAM-v1)

High beam describes the customer-visible behaviour for HIGH_BEAM. On/off high beam control with stalk and basic anti-glare rules.

3.4.1 Identifier and overview

Function ID: CF-HIGH_BEAM-v1
Function code: HIGH_BEAM
Domain: lighting
Logical owner: **Exterior Light Control Team** (elc-team@example.com)

3.4.2 Scenario narrative

This function can be understood as a message journey through the vehicle: a request is detected, validated against enabling conditions, translated into lamp commands, and finally reported back to the driver via tell■tales and HMI status.

Involved ECUs (in typical order):

- Body Control Module ECU (BCM)
- Exterior Light Control ECU (ELC)
- Headlamp Control Unit ECU (HCU)
- Instrument Cluster ECU (IC)

3.4.3 Operating conditions

- Vehicle electrical system shall be in a driving-capable or awake state (ignition/ready or permitted wake-up).
- Signals required for decision-making shall be valid (no stale timestamps, no implausible values).
- If a global lighting power limitation is active (POWER_LIMIT), optional functions shall be inhibited or degraded according to configuration.
- A driver request shall be present via stalk/switch input or configured automatic mode.

3.4.4 Functional behaviour

- High beam is activated on driver request and shall command the headlamp unit to switch to high-beam intensity. The tell■tale shall inform the driver that high beam is active.
- If the system is in power-limit or a blocking mode, high beam may be inhibited and the tell■tale shall reflect the final output state.

3.4.5 Normative requirements

- **REQ-HIGH_BEAM-001** — The function shall evaluate all operating conditions before activating any lamp output.
- **REQ-HIGH_BEAM-002** — The function shall accept driver/input requests via the representative input signals listed in the interfaces section.
- **REQ-HIGH_BEAM-004** — The function shall translate the decision into lamp command outputs to the applicable lamp driver ECUs.
- **REQ-HIGH_BEAM-005** — The function shall report status to HMI, including tell■tales where applicable.

3.4.6 Interfaces and signal exchange

Representative request / driver input signals:

- `STALK_HIGH_BEAM` — High beam stalk request (BCM→ELC)

Representative lamp command signals:

- `CMD_HIGH_BEAM` — Command high beam (ELC→HCU)

Representative HMI / status signals:

- `TT_HIGH_BEAM` — Tell-tale high beam (ELC→IC)
- `HMI_LIGHT_STATUS` — Lighting status to infotainment (ELC→IHU)

Representative signal path (hop-by-hop):

- 1. `BCM` → `ELC` via `CAN_BODY` (CAN, 10 ms)
- 2. `ELC` → `HCU` via `LIN_FRONT` (LIN, 15 ms)
- 3. `ELC` → `IC` via `CAN_BODY` (CAN, 20 ms)

3.4.7 Diagnostics and fault handling

If required inputs are invalid, the function shall fail safe (no unintended lamp activation) and expose a diagnostic state via HMI/status reporting. If a commanded lamp driver ECU does not acknowledge a request (bus off, LIN timeout), the function shall continue operating with remaining available outputs and log a fault condition.

3.4.8 Acceptance criteria (example)

- Given valid enabling conditions, the commanded lamps reach the expected on/off pattern within the indicative latency budget.
- Given a removed request or disabled condition, lamps return to OFF within the configured exit timing.
- Given an active function, HMI state matches lamp output state (no contradiction between tell■tale and lamps).

3.5 Main light (low beam + tail) (CF-MAIN_LIGHT-v1)

Main light (low beam + tail) describes the customer-visible behaviour for `MAIN_LIGHT`. Base low-beam and tail light function for night-time driving.

3.5.1 Identifier and overview

Function ID: `CF-MAIN_LIGHT-v1`
Function code: `MAIN_LIGHT`
Domain: lighting
Logical owner: **Exterior Light Control Team** (elc-team@example.com)

3.5.2 Scenario narrative

This function can be understood as a message journey through the vehicle: a request is detected, validated against enabling conditions, translated into lamp commands, and finally reported back to the driver via tell■tales and HMI status.

Involved ECUs (in typical order):

- Body Control Module ECU (`BCM`)
- Exterior Light Control ECU (`ELC`)
- Headlamp Control Unit ECU (`HCU`)
- Rear Light Control Unit ECU (`RLCU`)
- Instrument Cluster ECU (`IC`)

3.5.3 Operating conditions

- Vehicle electrical system shall be in a driving-capable or awake state (ignition/ready or permitted wake-up).
- Signals required for decision-making shall be valid (no stale timestamps, no implausible values).
- If a global lighting power limitation is active (`POWER_LIMIT`), optional functions shall be inhibited or degraded according to configuration.
- A driver request shall be present via stalk/switch input or configured automatic mode.

3.5.4 Functional behaviour

- Main light (low beam + tail lamps) provides the primary night-driving illumination. It can be activated by driver request or by an automatic mode depending on ambient light.
- When main light is on, low beam and tail lamps shall be commanded to their target intensities.

3.5.5 Normative requirements

- **REQ-MAIN_LIGHT-001** — The function shall evaluate all operating conditions before activating any lamp output.
- **REQ-MAIN_LIGHT-002** — The function shall accept driver/input requests via the representative input signals listed in the interfaces section.
- **REQ-MAIN_LIGHT-003** — The function shall use vehicle-state/sensor inputs (e.g., speed, brake pressure, ambient light) to gate or adapt behaviour.
- **REQ-MAIN_LIGHT-004** — The function shall translate the decision into lamp command outputs to the applicable lamp driver ECUs.

- **REQ-MAIN_LIGHT-005** — The function shall report status to HMI, including tell■tales where applicable.

3.5.6 Interfaces and signal exchange

Representative request / driver input signals:

- `STALK_PARK_LIGHT` — Park light request (BCM→ELC)
- `STALK_POSITION` — Position / side light request (BCM→ELC)

Representative state / sensor / power signals:

- `AMBIENT_LIGHT` — Ambient light level (ADAS→ELC)

Representative lamp command signals:

- `CMD_MAIN_LOWBEAM` — Command low beam (ELC→HCU)
- `CMD_TAIL_LIGHT` — Command tail lights (ELC→RLCU)

Representative HMI / status signals:

- `HMI_LIGHT_STATUS` — Lighting status to infotainment (ELC→IHU)

Representative signal path (hop-by-hop):

- 1. `BCM` → `ELC` via `CAN_BODY` (CAN, 10 ms)
- 2. `ELC` → `HCU` via `LIN_FRONT` (LIN, 15 ms)
- 3. `ELC` → `RLCU` via `LIN_REAR` (LIN, 15 ms)
- 4. `ELC` → `IC` via `CAN_BODY` (CAN, 20 ms)

3.5.7 Diagnostics and fault handling

If required inputs are invalid, the function shall fail safe (no unintended lamp activation) and expose a diagnostic state via HMI/status reporting. If a commanded lamp driver ECU does not acknowledge a request (bus off, LIN timeout), the function shall continue operating with remaining available outputs and log a fault condition.

3.5.8 Acceptance criteria (example)

- Given valid enabling conditions, the commanded lamps reach the expected on/off pattern within the indicative latency budget.
- Given a removed request or disabled condition, lamps return to OFF within the configured exit timing.
- Given an active function, HMI state matches lamp output state (no contradiction between tell■tale and lamps).

3.6 Park light (CF-PARK_LIGHT-v1)

Park light describes the customer-visible behaviour for `PARK_LIGHT`. Parking light function for stationary vehicle signaling.

3.6.1 Identifier and overview

Function ID: `CF-PARK_LIGHT-v1`
Function code: `PARK_LIGHT`
Domain: lighting

Logical owner: **Exterior Light Control Team** (elc-team@example.com)

3.6.2 Scenario narrative

This function can be understood as a message journey through the vehicle: a request is detected, validated against enabling conditions, translated into lamp commands, and finally reported back to the driver via tell■tales and HMI status.

Involved ECUs (in typical order):

- Body Control Module ECU (`BCM`)
- Exterior Light Control ECU (`ELC`)
- Rear Light Control Unit ECU (`RLCU`)
- Instrument Cluster ECU (`IC`)

3.6.3 Operating conditions

- Vehicle electrical system shall be in a driving-capable or awake state (ignition/ready or permitted wake-up).
- Signals required for decision-making shall be valid (no stale timestamps, no implausible values).
- If a global lighting power limitation is active (`POWER_LIMIT`), optional functions shall be inhibited or degraded according to configuration.
- A driver request shall be present via stalk/switch input or configured automatic mode.

3.6.4 Functional behaviour

- Park light provides a low-intensity stationary lighting mode for parked vehicles. When requested, the vehicle shall activate the configured park/position/tail lamps while respecting battery and power constraints.

3.6.5 Normative requirements

- **REQ-PARK_LIGHT-001** — The function shall evaluate all operating conditions before activating any lamp output.
- **REQ-PARK_LIGHT-002** — The function shall accept driver/input requests via the representative input signals listed in the interfaces section.
- **REQ-PARK_LIGHT-004** — The function shall translate the decision into lamp command outputs to the applicable lamp driver ECUs.
- **REQ-PARK_LIGHT-005** — The function shall report status to HMI, including tell■tales where applicable.

3.6.6 Interfaces and signal exchange

Representative request / driver input signals:

- `STALK_PARK_LIGHT` — Park light request (BCM→ELC)

Representative lamp command signals:

- `CMD_TAIL_LIGHT` — Command tail lights (ELC→RLCU)

Representative HMI / status signals:

- `HMI_LIGHT_STATUS` — Lighting status to infotainment (ELC→IHU)

Representative signal path (hop-by-hop):

- 1. BCM → ELC via CAN_BODY (CAN, 10 ms)
- 2. ELC → RLCU via LIN_REAR (LIN, 15 ms)
- 3. ELC → IC via CAN_BODY (CAN, 20 ms)

3.6.7 Diagnostics and fault handling

If required inputs are invalid, the function shall fail safe (no unintended lamp activation) and expose a diagnostic state via HMI/status reporting. If a commanded lamp driver ECU does not acknowledge a request (bus off, LIN timeout), the function shall continue operating with remaining available outputs and log a fault condition.

3.6.8 Acceptance criteria (example)

- Given valid enabling conditions, the commanded lamps reach the expected on/off pattern within the indicative latency budget.
- Given a removed request or disabled condition, lamps return to OFF within the configured exit timing.
- Given an active function, HMI state matches lamp output state (no contradiction between tell■tale and lamps).

3.7 Position / side light (CF-POSITION_LIGHT-v1)

Position / side light describes the customer-visible behaviour for POSITION_LIGHT. Side/position light function for vehicle contour visibility.

3.7.1 Identifier and overview

Function ID: CF-POSITION_LIGHT-v1
Function code: POSITION_LIGHT
Domain: lighting
Logical owner: **Exterior Light Control Team** (elc-team@example.com)

3.7.2 Scenario narrative

This function can be understood as a message journey through the vehicle: a request is detected, validated against enabling conditions, translated into lamp commands, and finally reported back to the driver via tell■tales and HMI status.

Involved ECUs (in typical order):

- Body Control Module ECU (BCM)
- Exterior Light Control ECU (ELC)
- Headlamp Control Unit ECU (HCU)
- Rear Light Control Unit ECU (RLCU)

3.7.3 Operating conditions

- Vehicle electrical system shall be in a driving-capable or awake state (ignition/ready or permitted wake-up).
- Signals required for decision-making shall be valid (no stale timestamps, no implausible values).

- If a global lighting power limitation is active (`POWER_LIMIT`), optional functions shall be inhibited or degraded according to configuration.
- A driver request shall be present via stalk/switch input or configured automatic mode.

3.7.4 Functional behaviour

- Position/side lights define the vehicle outline and are typically used as a base lighting mode. On request, the system shall turn on the appropriate position lamps and tail lamps as configured.

3.7.5 Normative requirements

- **REQ-POSITION_LIGHT-001** — The function shall evaluate all operating conditions before activating any lamp output.
- **REQ-POSITION_LIGHT-002** — The function shall accept driver/input requests via the representative input signals listed in the interfaces section.
- **REQ-POSITION_LIGHT-004** — The function shall translate the decision into lamp command outputs to the applicable lamp driver ECUs.
- **REQ-POSITION_LIGHT-005** — The function shall report status to HMI, including tell■tales where applicable.

3.7.6 Interfaces and signal exchange

Representative request / driver input signals:

- `STALK_POSITION` — Position / side light request (BCM→ELC)

Representative lamp command signals:

- `CMD_TAIL_LIGHT` — Command tail lights (ELC→RLCU)

Representative HMI / status signals:

- `HMI_LIGHT_STATUS` — Lighting status to infotainment (ELC→IHU)

Representative signal path (hop-by-hop):

- 1. `BCM` → `ELC` via `CAN_BODY` (CAN, 10 ms)
- 2. `ELC` → `HCU` via `LIN_FRONT` (LIN, 15 ms)
- 3. `ELC` → `RLCU` via `LIN_REAR` (LIN, 15 ms)

3.7.7 Diagnostics and fault handling

If required inputs are invalid, the function shall fail safe (no unintended lamp activation) and expose a diagnostic state via HMI/status reporting. If a commanded lamp driver ECU does not acknowledge a request (bus off, LIN timeout), the function shall continue operating with remaining available outputs and log a fault condition.

3.7.8 Acceptance criteria (example)

- Given valid enabling conditions, the commanded lamps reach the expected on/off pattern within the indicative latency budget.
- Given a removed request or disabled condition, lamps return to OFF within the configured exit timing.
- Given an active function, HMI state matches lamp output state (no contradiction between tell■tale and lamps).

3.8 Daytime Running Light (DRL) (CF-DRL-v1)

Daytime Running Light (DRL) describes the customer-visible behaviour for `DRL`. Front daytime running lights for improved daytime conspicuity.

3.8.1 Identifier and overview

Function ID: `CF-DRL-v1`
Function code: `DRL`
Domain: lighting
Logical owner: **Exterior Light Control Team** (elc-team@example.com)

3.8.2 Scenario narrative

This function can be understood as a message journey through the vehicle: a request is detected, validated against enabling conditions, translated into lamp commands, and finally reported back to the driver via tell■tales and HMI status.

Involved ECUs (in typical order):

- Body Control Module ECU (`BCM`)
- Exterior Light Control ECU (`ELC`)
- Headlamp Control Unit ECU (`HCU`)
- Instrument Cluster ECU (`IC`)
- Infotainment Head Unit ECU (`IHU`)

3.8.3 Operating conditions

- Vehicle electrical system shall be in a driving-capable or awake state (ignition/ready or permitted wake-up).
- Signals required for decision-making shall be valid (no stale timestamps, no implausible values).
- If a global lighting power limitation is active (`POWER_LIMIT`), optional functions shall be inhibited or degraded according to configuration.
- A driver request shall be present via stalk/switch input or configured automatic mode.

3.8.4 Functional behaviour

- Daytime Running Lights increase vehicle conspicuity during daytime driving. When enabled, DRL shall be active according to vehicle state and ambient light conditions, and shall be deactivated when low beam is required.

3.8.5 Normative requirements

- **REQ-DRL-001** — The function shall evaluate all operating conditions before activating any lamp output.
- **REQ-DRL-002** — The function shall accept driver/input requests via the representative input signals listed in the interfaces section.
- **REQ-DRL-003** — The function shall use vehicle-state/sensor inputs (e.g., speed, brake pressure, ambient light) to gate or adapt behaviour.
- **REQ-DRL-004** — The function shall translate the decision into lamp command outputs to the applicable lamp driver ECUs.
- **REQ-DRL-005** — The function shall report status to HMI, including tell■tales where applicable.

3.8.6 Interfaces and signal exchange

Representative request / driver input signals:

- `STALK_DRL_MODE` — DRL enable request (BCM→ELC)

Representative state / sensor / power signals:

- `VEH_SPEED` — Vehicle speed (ESP_BRS→ELC)
- `AMBIENT_LIGHT` — Ambient light level (ADAS→ELC)

Representative lamp command signals:

- `CMD_DRL` — Command DRL (ELC→HCU)

Representative HMI / status signals:

- `HMI_LIGHT_STATUS` — Lighting status to infotainment (ELC→IHU)

Representative signal path (hop-by-hop):

- 1. `BCM` → `ELC` via `CAN_BODY` (CAN, 10 ms)
- 2. `ELC` → `HCU` via `LIN_FRONT` (LIN, 15 ms)
- 3. `ELC` → `IC` via `CAN_BODY` (CAN, 20 ms)
- 4. `ELC` → `IHU` via `ETH_BACKBONE` (ETHERNET, 30 ms)

3.8.7 Diagnostics and fault handling

If required inputs are invalid, the function shall fail safe (no unintended lamp activation) and expose a diagnostic state via HMI/status reporting. If a commanded lamp driver ECU does not acknowledge a request (bus off, LIN timeout), the function shall continue operating with remaining available outputs and log a fault condition.

3.8.8 Acceptance criteria (example)

- Given valid enabling conditions, the commanded lamps reach the expected on/off pattern within the indicative latency budget.
- Given a removed request or disabled condition, lamps return to OFF within the configured exit timing.
- Given an active function, HMI state matches lamp output state (no contradiction between tell■tale and lamps).

3.9 Front fog light (CF-FRONT_FOG-v1)

Front fog light describes the customer-visible behaviour for `FRONT_FOG`. Front fog lamp function for fog or heavy rain conditions.

3.9.1 Identifier and overview

Function ID: `CF-FRONT_FOG-v1`
Function code: `FRONT_FOG`
Domain: lighting
Logical owner: **Exterior Light Control Team** (elc-team@example.com)

3.9.2 Scenario narrative

This function can be understood as a message journey through the vehicle: a request is detected, validated against enabling conditions, translated into lamp commands, and finally reported back to the driver via tell■tales and HMI status.

Involved ECUs (in typical order):

- Body Control Module ECU (BCM)
- Exterior Light Control ECU (ELC)
- Headlamp Control Unit ECU (HCU)
- Instrument Cluster ECU (IC)

3.9.3 Operating conditions

- Vehicle electrical system shall be in a driving-capable or awake state (ignition/ready or permitted wake-up).
- Signals required for decision-making shall be valid (no stale timestamps, no implausible values).
- If a global lighting power limitation is active (POWER_LIMIT), optional functions shall be inhibited or degraded according to configuration.
- A driver request shall be present via stalk/switch input or configured automatic mode.

3.9.4 Functional behaviour

- Front fog lights improve near-field illumination in fog and precipitation. When the driver requests front fog, the system shall activate front fog lamps subject to prerequisites and shall provide status to HMI.

3.9.5 Normative requirements

- **REQ-FRONT_FOG-001** — The function shall evaluate all operating conditions before activating any lamp output.
- **REQ-FRONT_FOG-002** — The function shall accept driver/input requests via the representative input signals listed in the interfaces section.
- **REQ-FRONT_FOG-004** — The function shall translate the decision into lamp command outputs to the applicable lamp driver ECUs.
- **REQ-FRONT_FOG-005** — The function shall report status to HMI, including tell■tales where applicable.

3.9.6 Interfaces and signal exchange

Representative request / driver input signals:

- STALK_FOG_FRONT — Front fog request (BCM→ELC)

Representative lamp command signals:

- CMD_FRONT_FOG — Command front fog lamp (ELC→HCU)

Representative HMI / status signals:

- HMI_LIGHT_STATUS — Lighting status to infotainment (ELC→IHU)

Representative signal path (hop-by-hop):

- 1. BCM → ELC via CAN_BODY (CAN, 10 ms)
- 2. ELC → HCU via LIN_FRONT (LIN, 15 ms)

- 3. ELC → IC via CAN_BODY (CAN, 20 ms)

3.9.7 Diagnostics and fault handling

If required inputs are invalid, the function shall fail safe (no unintended lamp activation) and expose a diagnostic state via HMI/status reporting. If a commanded lamp driver ECU does not acknowledge a request (bus off, LIN timeout), the function shall continue operating with remaining available outputs and log a fault condition.

3.9.8 Acceptance criteria (example)

- Given valid enabling conditions, the commanded lamps reach the expected on/off pattern within the indicative latency budget.
- Given a removed request or disabled condition, lamps return to OFF within the configured exit timing.
- Given an active function, HMI state matches lamp output state (no contradiction between tell■tale and lamps).

3.10 Reverse light (CF-REVERSE_LIGHT-v1)

Reverse light describes the customer-visible behaviour for REVERSE_LIGHT. Rear reverse lamp function when vehicle is in reverse gear.

3.10.1 Identifier and overview

Function ID: CF-REVERSE_LIGHT-v1
Function code: REVERSE_LIGHT
Domain: lighting
Logical owner: **Exterior Light Control Team** (elc-team@example.com)

3.10.2 Scenario narrative

This function can be understood as a message journey through the vehicle: a request is detected, validated against enabling conditions, translated into lamp commands, and finally reported back to the driver via tell■tales and HMI status.

Involved ECUs (in typical order):

- Brake / ESP ECU ECU (ESP_BRS)
- Exterior Light Control ECU (ELC)
- Rear Light Control Unit ECU (RLCU)
- Surround View System ECU (SVS)
- Instrument Cluster ECU (IC)

3.10.3 Operating conditions

- Vehicle electrical system shall be in a driving-capable or awake state (ignition/ready or permitted wake-up).
- Signals required for decision-making shall be valid (no stale timestamps, no implausible values).
- If a global lighting power limitation is active (POWER_LIMIT), optional functions shall be inhibited or degraded according to configuration.
- Reverse gear shall be engaged (gear status indicates reverse).

3.10.4 Functional behaviour

- Reverse lamps provide rear illumination when reversing and shall be activated automatically when reverse gear is selected. Reverse lamps shall be deactivated when reverse gear is not selected.

3.10.5 Normative requirements

- **REQ-REVERSE_LIGHT-001** — The function shall evaluate all operating conditions before activating any lamp output.
- **REQ-REVERSE_LIGHT-003** — The function shall use vehicle-state/sensor inputs (e.g., speed, brake pressure, ambient light) to gate or adapt behaviour.
- **REQ-REVERSE_LIGHT-004** — The function shall translate the decision into lamp command outputs to the applicable lamp driver ECUs.
- **REQ-REVERSE_LIGHT-005** — The function shall report status to HMI, including tell■tales where applicable.

3.10.6 Interfaces and signal exchange

Representative state / sensor / power signals:

- `GEAR_REVERSE` — Reverse gear status (ESP_BRS→ELC)

Representative lamp command signals:

- `CMD_REVERSE_LIGHT` — Command reverse lamp (ELC→RLCU)

Representative HMI / status signals:

- `HMI_LIGHT_STATUS` — Lighting status to infotainment (ELC→IHU)

Representative signal path (hop-by-hop):

- 1. `ESP_BRS` → `ELC` via `CAN_CHASSIS` (CAN, 10 ms)
- 2. `ELC` → `RLCU` via `LIN_REAR` (LIN, 15 ms)
- 3. `ELC` → SVS via `CAN_BODY` (CAN, 20 ms)
- 4. `ELC` → `IC` via `CAN_BODY` (CAN, 20 ms)

3.10.7 Diagnostics and fault handling

If required inputs are invalid, the function shall fail safe (no unintended lamp activation) and expose a diagnostic state via HMI/status reporting. If a commanded lamp driver ECU does not acknowledge a request (bus off, LIN timeout), the function shall continue operating with remaining available outputs and log a fault condition.

3.10.8 Acceptance criteria (example)

- Given valid enabling conditions, the commanded lamps reach the expected on/off pattern within the indicative latency budget.
- Given a removed request or disabled condition, lamps return to OFF within the configured exit timing.
- Given an active function, HMI state matches lamp output state (no contradiction between tell■tale and lamps).

3.11 Cornering light (CF-CORNERING_LIGHT-v1)

Cornering light describes the customer-visible behaviour for `CORNERING_LIGHT`. Cornering light to illuminate inside of turn at low speed.

3.11.1 Identifier and overview

Function ID: `CF-CORNERING_LIGHT-v1`
Function code: `CORNERING_LIGHT`
Domain: lighting
Logical owner: **Light System Master Team** (lsm-team@example.com)

3.11.2 Scenario narrative

This function can be understood as a message journey through the vehicle: a request is detected, validated against enabling conditions, translated into lamp commands, and finally reported back to the driver via tell■tales and HMI status.

Involved ECUs (in typical order):

- Brake / ESP ECU ECU (`ESP_BRS`)
- Exterior Light Control ECU (`ELC`)
- ADAS Domain Controller ECU (`ADAS`)
- Light System Master ECU (`LSM`)
- Headlamp Control Unit ECU (`HCU`)
- Instrument Cluster ECU (`IC`)

3.11.3 Operating conditions

- Vehicle electrical system shall be in a driving-capable or awake state (ignition/ready or permitted wake-up).
- Signals required for decision-making shall be valid (no stale timestamps, no implausible values).
- If a global lighting power limitation is active (`POWER_LIMIT`), optional functions shall be inhibited or degraded according to configuration.
- Steering angle and speed shall be within the configured window that enables cornering assistance.

3.11.4 Functional behaviour

- Cornering light provides additional lateral illumination during low-speed turns. When steering angle exceeds a threshold at eligible speeds, the system shall activate the corresponding cornering lamp.
- Activation and deactivation shall be smooth (ramp up/down) to avoid visually abrupt light changes.

3.11.5 Normative requirements

- **REQ-CORNERING_LIGHT-001** — The function shall evaluate all operating conditions before activating any lamp output.
- **REQ-CORNERING_LIGHT-003** — The function shall use vehicle-state/sensor inputs (e.g., speed, brake pressure, ambient light) to gate or adapt behaviour.

- **REQ-CORNERING_LIGHT-004** — The function shall translate the decision into lamp command outputs to the applicable lamp driver ECUs.
- **REQ-CORNERING_LIGHT-005** — The function shall report status to HMI, including tell■tales where applicable.

3.11.6 Interfaces and signal exchange

Representative state / sensor / power signals:

- `VEH_SPEED` — Vehicle speed (ESP_BRS→ELC)
- `STEERING_ANGLE` — Steering angle (ADAS→ELC)

Representative lamp command signals:

- `CMD_CORNERING_LEFT` — Command cornering lamp left (LSM→HCU)

Representative HMI / status signals:

- `HMI_LIGHT_STATUS` — Lighting status to infotainment (ELC→IHU)

Representative signal path (hop-by-hop):

- 1. `ESP_BRS` → `ELC` via `CAN_CHASSIS` (CAN, 10 ms)
- 2. `ADAS` → `ELC` via `CAN_CHASSIS` (CAN, 10 ms)
- 3. `ELC` → `LSM` via `CAN_BODY` (CAN, 10 ms)
- 4. `LSM` → `HCU` via `LIN_FRONT` (LIN, 15 ms)
- 5. `LSM` → `IC` via `CAN_BODY` (CAN, 20 ms)

3.11.7 Diagnostics and fault handling

If required inputs are invalid, the function shall fail safe (no unintended lamp activation) and expose a diagnostic state via HMI/status reporting. If a commanded lamp driver ECU does not acknowledge a request (bus off, LIN timeout), the function shall continue operating with remaining available outputs and log a fault condition.

3.11.8 Acceptance criteria (example)

- Given valid enabling conditions, the commanded lamps reach the expected on/off pattern within the indicative latency budget.
- Given a removed request or disabled condition, lamps return to OFF within the configured exit timing.
- Given an active function, HMI state matches lamp output state (no contradiction between tell■tale and lamps).

3.12 Adaptive high beam (CF-ADAPTIVE_HIGH_BEAM-v1)

Adaptive high beam describes the customer-visible behaviour for `ADAPTIVE_HIGH_BEAM`. Adaptive high beam with camera-based glare avoidance.

3.12.1 Identifier and overview

Function ID: `CF-ADAPTIVE_HIGH_BEAM-v1`
Function code: `ADAPTIVE_HIGH_BEAM`
Domain: lighting

Logical owner: **Light System Master Team** (lsm-team@example.com)

3.12.2 Scenario narrative

This function can be understood as a message journey through the vehicle: a request is detected, validated against enabling conditions, translated into lamp commands, and finally reported back to the driver via tell■tales and HMI status.

Involved ECUs (in typical order):

- ADAS Domain Controller ECU (`ADAS`)
- Light System Master ECU (`LSM`)
- Headlamp Control Unit ECU (`HCU`)
- Instrument Cluster ECU (`IC`)
- Infotainment Head Unit ECU (`IHU`)

3.12.3 Operating conditions

- Vehicle electrical system shall be in a driving-capable or awake state (ignition/ready or permitted wake-up).
- Signals required for decision-making shall be valid (no stale timestamps, no implausible values).
- If a global lighting power limitation is active (`POWER_LIMIT`), optional functions shall be inhibited or degraded according to configuration.
- High-beam assistant shall be enabled by the driver and not blocked by sensor availability.

3.12.4 Functional behaviour

- Adaptive high beam assists the driver by automatically toggling high beam in suitable conditions. The system shall consider ambient light and driving context and shall fall back to standard high beam behaviour when assistant functions are unavailable.

3.12.5 Normative requirements

- **REQ-ADAPTIVE_HIGH_BEAM-001** — The function shall evaluate all operating conditions before activating any lamp output.
- **REQ-ADAPTIVE_HIGH_BEAM-002** — The function shall accept driver/input requests via the representative input signals listed in the interfaces section.
- **REQ-ADAPTIVE_HIGH_BEAM-003** — The function shall use vehicle-state/sensor inputs (e.g., speed, brake pressure, ambient light) to gate or adapt behaviour.
- **REQ-ADAPTIVE_HIGH_BEAM-004** — The function shall translate the decision into lamp command outputs to the applicable lamp driver ECUs.
- **REQ-ADAPTIVE_HIGH_BEAM-005** — The function shall report status to HMI, including tell■tales where applicable.

3.12.6 Interfaces and signal exchange

Representative request / driver input signals:

- `STALK_HIGH_BEAM` — High beam stalk request (BCM→ELC)

Representative state / sensor / power signals:

- VEH_SPEED — Vehicle speed (ESP_BRS→ELC)
- STEERING_ANGLE — Steering angle (ADAS→ELC)
- AMBIENT_LIGHT — Ambient light level (ADAS→ELC)

Representative lamp command signals:

- CMD_HIGH_BEAM — Command high beam (ELC→HCU)

Representative HMI / status signals:

- TT_HIGH_BEAM — Tell-tale high beam (ELC→IC)
- HMI_LIGHT_STATUS — Lighting status to infotainment (ELC→IHU)

Representative signal path (hop-by-hop):

- 1. ADAS → LSM via CAN_CHASSIS (CAN, 10 ms)
- 2. LSM → HCU via LIN_FRONT (LIN, 15 ms)
- 3. LSM → IC via CAN_BODY (CAN, 20 ms)
- 4. LSM → IHU via ETH_BACKBONE (ETHERNET, 30 ms)

3.12.7 Diagnostics and fault handling

If required inputs are invalid, the function shall fail safe (no unintended lamp activation) and expose a diagnostic state via HMI/status reporting. If a commanded lamp driver ECU does not acknowledge a request (bus off, LIN timeout), the function shall continue operating with remaining available outputs and log a fault condition.

3.12.8 Acceptance criteria (example)

- Given valid enabling conditions, the commanded lamps reach the expected on/off pattern within the indicative latency budget.
- Given a removed request or disabled condition, lamps return to OFF within the configured exit timing.
- Given an active function, HMI state matches lamp output state (no contradiction between tell■tale and lamps).

3.13 Hazard warning after emergency brake (CF-EMERGENCY_HAZARD-v1)

Hazard warning after emergency brake describes the customer-visible behaviour for EMERGENCY_HAZARD. Automatic hazard warning activation after emergency braking.

3.13.1 Identifier and overview

Function ID: CF-EMERGENCY_HAZARD-v1
Function code: EMERGENCY_HAZARD
Domain: lighting
Logical owner: **Light System Master Team** (lsm-team@example.com)

3.13.2 Scenario narrative

This function can be understood as a message journey through the vehicle: a request is detected, validated against enabling conditions, translated into lamp commands, and finally reported back to

the driver via tell■tales and HMI status.

Involved ECUs (in typical order):

- Brake / ESP ECU ECU (`ESP_BRS`)
- Light System Master ECU (`LSM`)
- Headlamp Control Unit ECU (`HCU`)
- Rear Light Control Unit ECU (`RLCU`)
- Instrument Cluster ECU (`IC`)
- Telematics ECU ECU (`TELE`)

3.13.3 Operating conditions

- Vehicle electrical system shall be in a driving-capable or awake state (ignition/ready or permitted wake-up).
- Signals required for decision-making shall be valid (no stale timestamps, no implausible values).
- If a global lighting power limitation is active (`POWER_LIMIT`), optional functions shall be inhibited or degraded according to configuration.
- An emergency braking event shall be detected (high deceleration / brake pressure while moving).

3.13.4 Functional behaviour

- During emergency braking, hazard warning shall be activated automatically to alert surrounding traffic. The hazard pattern shall be broadcast to front and rear lamps and a hazard tell■tale shall be displayed.
- The hazard warning shall remain active for a configured duration or until the driver cancels it, depending on vehicle configuration.

3.13.5 Normative requirements

- **REQ-EMERGENCY_HAZARD-001** — The function shall evaluate all operating conditions before activating any lamp output.
- **REQ-EMERGENCY_HAZARD-002** — The function shall accept driver/input requests via the representative input signals listed in the interfaces section.
- **REQ-EMERGENCY_HAZARD-003** — The function shall use vehicle-state/sensor inputs (e.g., speed, brake pressure, ambient light) to gate or adapt behaviour.
- **REQ-EMERGENCY_HAZARD-004** — The function shall translate the decision into lamp command outputs to the applicable lamp driver ECUs.
- **REQ-EMERGENCY_HAZARD-005** — The function shall report status to HMI, including tell■tales where applicable.
- **REQ-EMERGENCY_HAZARD-006** — The function shall provide events/status suitable for telematics logging when telematics is part of the defined signal path.

3.13.6 Interfaces and signal exchange

Representative request / driver input signals:

- `HAZARD_SWITCH` — Hazard switch status (BCM→LSM)

Representative state / sensor / power signals:

- BRAKE_PRESSURE — Brake pressure / decel (ESP_BRS→ELC)
- VEH_SPEED — Vehicle speed (ESP_BRS→ELC)

Representative lamp command signals:

- CMD_HAZARD_ALL — Command hazard pattern all lamps (LSM→RLCU)

Representative HMI / status signals:

- TT_HAZARD — Tell-tale hazard (LSM→IC)

Representative signal path (hop-by-hop):

- 1. ESP_BRS → LSM via CAN_CHASSIS (CAN, 10 ms)
- 2. LSM → HCU via LIN_FRONT (LIN, 15 ms)
- 3. LSM → RLCU via LIN_REAR (LIN, 15 ms)
- 4. LSM → IC via CAN_BODY (CAN, 20 ms)
- 5. LSM → TELE via ETH_BACKBONE (ETHERNET, 30 ms)

3.13.7 Diagnostics and fault handling

If required inputs are invalid, the function shall fail safe (no unintended lamp activation) and expose a diagnostic state via HMI/status reporting. If a commanded lamp driver ECU does not acknowledge a request (bus off, LIN timeout), the function shall continue operating with remaining available outputs and log a fault condition.

3.13.8 Acceptance criteria (example)

- Given valid enabling conditions, the commanded lamps reach the expected on/off pattern within the indicative latency budget.
- Given a removed request or disabled condition, lamps return to OFF within the configured exit timing.
- Given an active function, HMI state matches lamp output state (no contradiction between tell■tale and lamps).

3.14 Welcome / goodbye lighting (CF-WELCOME_GOODBYE-v1)

Welcome / goodbye lighting describes the customer-visible behaviour for WELCOME_GOODBYE. Inszenierungs-/welcome/goodbye lighting sequences at lock/unlock.

3.14.1 Identifier and overview

Function ID: CF-WELCOME_GOODBYE-v1
Function code: WELCOME_GOODBYE
Domain: lighting
Logical owner: **Light System Master Team** (lsm-team@example.com)

3.14.2 Scenario narrative

This function can be understood as a message journey through the vehicle: a request is detected, validated against enabling conditions, translated into lamp commands, and finally reported back to the driver via tell■tales and HMI status.

Involved ECUs (in typical order):

- Body Control Module ECU (BCM)
- Light System Master ECU (LSM)
- Headlamp Control Unit ECU (HCU)
- Rear Light Control Unit ECU (RLCU)
- Door Light Controller FL ECU (DLC_FL)
- Door Light Controller FR ECU (DLC_FR)
- Interior Light Controller ECU (ILC)
- Instrument Cluster ECU (IC)
- Infotainment Head Unit ECU (IHU)
- Telematics ECU ECU (TELE)

3.14.3 Operating conditions

- Vehicle electrical system shall be in a driving-capable or awake state (ignition/ready or permitted wake-up).
- Signals required for decision-making shall be valid (no stale timestamps, no implausible values).
- If a global lighting power limitation is active (POWER_LIMIT), optional functions shall be inhibited or degraded according to configuration.
- A lock/unlock or approach/leave event shall be detected and permitted by comfort settings.

3.14.4 Functional behaviour

- Welcome/goodbye lighting provides a comfort experience by illuminating selected exterior and door/mirror lamps on lock/unlock events. The system shall choreograph a timed sequence and provide HMI feedback as configured.
- The sequence may also coordinate with interior lighting and may generate an event for telematics logging.

3.14.5 Normative requirements

- **REQ-WELCOME_GOODBYE-001** — The function shall evaluate all operating conditions before activating any lamp output.
- **REQ-WELCOME_GOODBYE-002** — The function shall accept driver/input requests via the representative input signals listed in the interfaces section.
- **REQ-WELCOME_GOODBYE-004** — The function shall translate the decision into lamp command outputs to the applicable lamp driver ECUs.
- **REQ-WELCOME_GOODBYE-005** — The function shall report status to HMI, including tell■tales where applicable.
- **REQ-WELCOME_GOODBYE-006** — The function shall provide events/status suitable for telematics logging when telematics is part of the defined signal path.

3.14.6 Interfaces and signal exchange

Representative request / driver input signals:

- REMOTE_UNLOCK — Remote unlock event (BCM→LSM)
- REMOTE_LOCK — Remote lock event (BCM→LSM)

Representative lamp command signals:

- CMD_WELCOME_EXT — Command welcome exterior lamps (LSM→HCU)
- CMD_WELCOME_DOOR — Command door/mirror welcome lights (LSM→DLC_FL)
- CMD_WELCOME_DOOR_FR — Command door/mirror welcome lights FR (LSM→DLC_FR)

Representative HMI / status signals:

- HMI_LIGHT_STATUS — Lighting status to infotainment (ELC→IHU)

Representative signal path (hop-by-hop):

- 1. BCM → LSM via CAN_BODY (CAN, 10 ms)
- 2. LSM → HCU via LIN_FRONT (LIN, 15 ms)
- 3. LSM → RLCU via LIN_REAR (LIN, 15 ms)
- 4. LSM → DLC_FL via LIN_FRONT (LIN, 15 ms)
- 5. LSM → DLC_FR via LIN_FRONT (LIN, 15 ms)
- 6. LSM → ILC via CAN_BODY (CAN, 20 ms)
- 7. LSM → IC via CAN_BODY (CAN, 20 ms)
- 8. LSM → IHU via ETH_BACKBONE (ETHERNET, 30 ms)
- 9. LSM → TELE via ETH_BACKBONE (ETHERNET, 30 ms)

3.14.7 Diagnostics and fault handling

If required inputs are invalid, the function shall fail safe (no unintended lamp activation) and expose a diagnostic state via HMI/status reporting. If a commanded lamp driver ECU does not acknowledge a request (bus off, LIN timeout), the function shall continue operating with remaining available outputs and log a fault condition.

3.14.8 Acceptance criteria (example)

- Given valid enabling conditions, the commanded lamps reach the expected on/off pattern within the indicative latency budget.
- Given a removed request or disabled condition, lamps return to OFF within the configured exit timing.
- Given an active function, HMI state matches lamp output state (no contradiction between tell■tale and lamps).

Appendix A — Interface catalogue (codes)

This appendix lists the representative signals used across the functions. It is intentionally compact and structured for parsing.

ADAS → ELC

- `AMBIENT_LIGHT` — Ambient light level [sensor]
- `STEERING_ANGLE` — Steering angle [vehicle_state]

ADAS → LSM

- `CAMERA_OBJECTS` — Camera objects for AHB [sensor]

BCM → ELC

- `STALK_DRL_MODE` — DRL enable request [driver_input]
- `STALK_FOG_FRONT` — Front fog request [driver_input]
- `STALK_FOG_REAR` — Rear fog request [driver_input]
- `STALK_HIGH_BEAM` — High beam stalk request [driver_input]
- `STALK_PARK_LIGHT` — Park light request [driver_input]
- `STALK_POSITION` — Position / side light request [driver_input]
- `STALK_TURN_LEFT` — Left turn stalk request [driver_input]

BCM → LSM

- `HAZARD_SWITCH` — Hazard switch status [driver_input]
- `REMOTE_LOCK` — Remote lock event [driver_input]
- `REMOTE_UNLOCK` — Remote unlock event [driver_input]

ELC → HCU

- `CMD_BLINKER_LEFT_FRONT` — Command left front indicator [command]
- `CMD_DRL` — Command DRL [command]
- `CMD_FRONT_FOG` — Command front fog lamp [command]
- `CMD_HIGH_BEAM` — Command high beam [command]
- `CMD_MAIN_LOWBEAM` — Command low beam [command]

ELC → IC

- `TT_BLINKER_LEFT` — Tell-tale left indicator [status]
- `TT_FOG_REAR` — Tell-tale rear fog [status]
- `TT_HIGH_BEAM` — Tell-tale high beam [status]

ELC → IHU

- `HMI_LIGHT_STATUS` — Lighting status to infotainment [status]

ELC → RLCU

- `CMD_BLINKER_LEFT_REAR` — Command left rear indicator [command]

- `CMD_BRAKE_LIGHT` — Command brake lights [command]
- `CMD_REAR_FOG` — Command rear fog lamp [command]
- `CMD_REVERSE_LIGHT` — Command reverse lamp [command]
- `CMD_TAIL_LIGHT` — Command tail lights [command]

ESP_BRS → ELC

- `BRAKE_PRESSURE` — Brake pressure / decel [vehicle_state]
- `GEAR_REVERSE` — Reverse gear status [vehicle_state]
- `VEH_SPEED` — Vehicle speed [vehicle_state]

LSM → DLC_FL

- `CMD_WELCOME_DOOR` — Command door/mirror welcome lights [command]

LSM → DLC_FR

- `CMD_WELCOME_DOOR_FR` — Command door/mirror welcome lights FR [command]

LSM → HCU

- `CMD_CORNERING_LEFT` — Command cornering lamp left [command]
- `CMD_WELCOME_EXT` — Command welcome exterior lamps [command]

LSM → IC

- `TT_HAZARD` — Tell-tale hazard [status]

LSM → RLCU

- `CMD_HAZARD_ALL` — Command hazard pattern all lamps [command]

PMU → ELC

- `POWER_LIMIT` — Lighting power limitation [power_state]

Appendix B — Glossary

- **ECU**: Electronic Control Unit. A computing node in the vehicle.
- **Bus**: Network connecting ECUs (CAN/LIN/Ethernet).
- **Signal**: A structured data element exchanged between ECUs (e.g., stalk request or lamp command).
- **Signal path**: A hop-by-hop chain showing how a function request or state flows through the vehicle networks.
- **Owner**: A responsible engineering team for a system or function.