**Multi Dispenser DLB Algorithm Working**

Summary

This document details the operational mechanics of the DLB algorithm, which will be at the core of the Multi Dispenser system. The algorithm features three distinct modes: Equal mode, FIFO mode, and Fast Lane mode.

Highlights

* Equal mode: The DLB algorithm prioritizes fair and balanced power distribution among all connected vehicles.
* FIFO mode: The DLB algorithm prioritizes vehicles based on their arrival order at the dispenser. The first vehicle in line gets served first, followed by the second, third, and so on. Power delivery follows this strict chronological order, regardless of the individual vehicle's demand or the number of connected vehicles.
* Fast Lane: The DLB algorithm considers one of the lanes as Fast Lane. This ensures that vehicles connected in this lane receive power quicker than those in the standard ones. While algorithm prioritizes the Fast Lane. Other lanes follow equal mode.

Working Example (360kW system (12 modules) with 4 Guns)

* Equal mode:
  + Example 1:
    - Car-1 needs 4-modules.
    - Car-2 needs 8-modules.
    - The algorithm would allocate:
      1. Car-1: 4-modules
      2. Car-2: 8-modules
  + Example 2:
    - Car-1 needs 2-modules.
    - Car-2 needs 4-modules.
    - Car-3 needs 6-modules.
    - The algorithm would allocate:
      1. Car-1: 2-modules
      2. Car-2: 4-modules
      3. Car-3: 6-modules
  + Example 3:
    - Car-1 needs 4-modules.
    - Car-2 needs 4-modules.
    - Car-3 needs 6-modules.
    - Car-4 needs 3-modules
    - The algorithm would allocate:
      1. Car-1: 3-modules
      2. Car-2: 3-modules
      3. Car-3: 3-modules
      4. Car-3: 3-modules
* FIFO mode:
  + Example 1:
    - Car-1 needs 4-modules.
    - Car-2 needs 8-modules.
    - The algorithm would allocate:
      1. Car-1: 4-modules
      2. Car-2: 8-modules
  + Example 2:
    - Car-1 needs 8-modules.
    - Car-2 needs 4-modules.
    - Car-3 needs 4-modules.
    - The algorithm would allocate:
      1. Car-1: 8-modules
      2. Car-2: 3-modules
      3. Car-3: 1-modules
  + Example 3:
    - Car-1 needs 8-modules.
    - Car-2 needs 4-modules.
    - Car-3 needs 4-modules.
    - Car-4 needs 3-modules
    - The algorithm would allocate:
      1. Car-1: 8-modules
      2. Car-2: 2-modules
      3. Car-3: 1-modules
      4. Car-3: 1-modules
* Fast Lane mode: (Need to define Fast Lane number, consider lane 1)