Rehydrator - CC1352R OAD Firmware Report

Yasin Engin | Embedded Firmware Case Study | May 18, 2026

Overview

Rehydrator is a Contiki-NG firmware adaptation for the TI CC1352R. The project focuses on a bootloader-compatible OAD layout where the TI BIM remains intact and old/new firmware images occupy predictable flash slots.

Problem

Firmware update work on a constrained MCU is unforgiving. The bootloader, CCFG area, OAD header, vector table, and image body all need to land at the expected addresses. A small mismatch can make an image unbootable or erase protected firmware state.

Architecture

The project keeps the TI BIM unchanged. The new firmware image is placed at page 0, the older firmware is preserved as a fallback image, and the BIM plus CCFG region remains protected at the end of flash.

Key Implementation Points

- C and Contiki-NG application paths for old-firmware and new-firmware builds.
- TI BIM/OAD image headers aligned with image type, entry point, and start address.
- Custom linker placement for old and new firmware slots.
- Makefile targets for upload-ready HEX and BIN artifacts.
- Verification notes for checking generated image placement before upload.

Memory Layout

- new-firmware: starts at 0x00000000 with entry around 0x00000100.
- old-firmware: preserved around 0x00030000 with entry around 0x00030100.
- BIM + CCFG: protected near 0x00056000 and should not be erased by application updates.

Lessons Learned

Bootloader compatibility is mostly about disciplined memory boundaries. Firmware artifacts need address-aware formats or very clear programming instructions. A small embedded project becomes much easier to review when linker decisions are documented next to the code.

Next Improvements

- Add automated section-address and slot-limit checks after each build.
- Document a release OAD flow with signed headers and CRC generation.
- Capture hardware test evidence for the old-to-new firmware transition.

Repository

https://github.com/YasinEnginn/rehydrator