{{{title}}}

U-blox memory issue impacting LTE devices (E Series LTE, Electron LTE, and Boron LTE) manufactured before October 2019.

Published: May 17, 2021

Issue summary

u-blox has informed Particle of an issue that affects SARA-R410M-02B modules running u-blox module firmware L0.0.00.00.05.06 version A.02.00.

These devices run a scheduled memory housekeeping procedure once every 124 days after the device's first registration with the network that takes 10-20 seconds to complete. If power to the device is interrupted or the device is power cycled during this procedure, it can cause memory corruption that permanently disrupts normal function of the modem and prevents it from booting or connecting to the network.

This issue primarily affects devices in low-power use cases (allowing their power sources to drain fully) and in use cases that frequently power-cycle the device. Devices that do not fully deplete their batteries or routinely experience quick, successive power-cycles have little to no risk of occurrence.

Products affected

The u-blox SARA-R410-02B is a component that is used in Particle E402 and Boron LTE devices.

The following Particle products with a manufacturing date before October 2019 are subject to the 124-day housekeeping issue:

- E Series LTE (E402)
- Electron LTE (ELC402)
- Boron LTE (BRN402)

The manufacturing date of a device can be identified by doing the following:

- Locate the serial number of the device by navigating to its device page in the Console, by using the [CLI command](#) `particle identify`, by checking your device's original packaging, or by reading the value from the u-blox module itself (Borons only).
  - Example: E42TAB852123456
- Characters 1-6 are for internal tracking, Character 7-9 identify the year and week of manufacturing.
  - This Electron in the example above was manufactured in 2018, week 52. This device would be subject to the 124-day housekeeping issue.

Issue mitigation

Risk of u-blox corruption is generally low given the infrequency of the housekeeping routine in question (10 seconds every 124 days). DeviceOS 1.3.1 and beyond have system firmware checks in place to address this issue in the course of normal operation. Keeping devices on up-to-date system firmware (at least DeviceOS 1.3.1) should significantly reduce exposure.

Updating affected devices

Particle firmware can be updated on affected devices using a standard OTA firmware update to Device OS [v1.1.1 / v1.2.1 / v1.3.1 or later](#) using the following guidelines:

- If running an older version of Device OS, please ensure that you test your application on the updated version of Device OS prior to deploying the update.
- During the update, ensure that power to the device is not interrupted and the device is not put to sleep.
- Ensure that the device is within cellular range to complete over the air update.

Mitigating higher-risk use cases

Care must be taken to avoid quick, successive modem power-cycling and/or manual modem hard resets during the housekeeping routine. Causes of potential exposure, alongside their respective mitigation practices, are enumerated below:

- **Low Battery States** - battery-powered devices that allow their batteries to drain completely (or near completely) are at *moderate* risk for u-blox corruption. If a device is powered on with minimal charge in its battery, and that battery is its sole power source, that battery may die during memory cleanup upon boot, interrupting the process and leading to corruption. It is worth stating that devices in a frequent low-power/brownout states are already vulnerable to serious performance issues - including degraded connectivity, flash corruption, keys issues, etc....

Mitigation Practice - devices powered by battery can monitor State of Charge via the FuelGauge Class of the DeviceOS API ([link](#)). Upon a given threshold (Particle recommends 20% SoC), devices can power themselves down safely - for example using our Hibernate Sleep API ([link](#)). It is recommended to keep the device powered down until charged to at least 30% SoC in order to add hysteresis and

ensure sufficient power during next boot, registration, and potential housekeeping.

- **Abrupt Power Removal** - devices without backup battery redundancy and connected only via VIN directly to a temporary power supply (e.g. vehicle) are at *moderate* risk for u-blox corruption under two conditions: a) if there is no way to qualify when power is removed from the device, b) if power is applied and removed frequently by design. Please note that a rapid succession (< 30 seconds) of hard modem power-cycles without any kind of buffering battery and/or supercapacitor could lead to u-blox corruption during this housekeeping period.

Mitigation Practice - using a backup battery or a supercapacitor that allow the modem at least 30 seconds of power upon boot will significantly decrease potential exposure to this issue.

- **Manual Modem Intervention** - manually hard-resetting the modem in user firmware (by writing to the u-blox modem reset pin) can present a *high* risk for u-blox corruption.

Mitigation Practice - hard resets in user firmware by writing to the modem reset pin should be avoided. It is also best to delay any other direct, manual intervention on the u-blox modem (e.g. direct read/write commands) until *at least* 30 seconds after boot (assuming 10 seconds for a successful registration). Please note that terms like "direct interventions" and "manual modem reset" do **not** refer to the standard functions of the DeviceOS API (e.g. `Cellular.off()`), but rather advanced commands not documented in our Reference Page.

Identifying impacted devices:

Particle devices with active u-blox memory corruption will fail to connect and can exhibit characteristic LED status signals and trace logs.

A Particle E Series LTE device in this state will blink its LED dark blue and will report:

INFO: SIM/modem not responsive or SIM not inserted/requires a PIN.

when producing logs from [Cloud Debug](#).

A Particle Boron LTE device in this state will blink its LED green and will report:

ERROR: No response from NCP

when producing logs from [Cloud Debug](#). *Please note that blinking green on a Particle Boron is a generic signal inclusive to all cellular connectivity problems - by no means should it be assumed that blinking green Borons have entered this state without the above trace log information.*