Release Notes 2.4.0

Table of Contents

Updates and new features

New features

MiraMesh

The MiraMesh network stack can now be used separately from MiraOS. This allows for integrating MiraMesh into other operating systems, such as FreeRTOS, or to run in minimal bare metal applications without an OS.

Universal Radio Protocol Time-Sharing

MiraOS and MiraMesh allows for the parallel integration of third party radio protocols, such as Thread or ZigBee. See documentation.

Link local multicast with user configurable performance

Link local multicast can now be used with greater speed and reduced risk of collisions than before. A reserved range of multicast groups can be used to control this. Note that link local multicast transmissions lack retransmissions and therefore have a lower delivery rate. See UDP API documentation.

Frontend support for Border Gateway

Border gateway now supports frontends, which is configured via the license. See documentation.

Reduced join times

The time it takes for a node to join a network has been roughly halved compared to MiraOS 2.3.1.

Configuration of root/mesh/leaf mode is done at runtime

The build target affixes "root", "denseroot", "mesh", "densemesh" have all been replaced with the "os" affix. Configuration of routing behaviour is set at network startup and cannot be changed once the network has been initialized.

Configurable neighbour table and link table

The link and neighbour tables sizes are now configurable at network startup. They are stored in a configurable memory area, easiest created by the MIRA_MEM_SET_BUFFER macro. See the provided examples.

Debug/log API

An API has been introduced to allow application code to receive events from Mira to facilitate debugging MiraOS.

Improvements

Safer storage of user config areas

User configuration is now stored in alternating flash areas, allowing for recovery to a previous version in the case of write failure.

Event timer resolution improvement

The previous resolution of 62.5ms has been changed to 1ms

Reduced hardware requirements

Mira now only uses RTC 2, leaving RTC 0 and 1 free for application use. Note that RTC 0 is still used by SoftDevice if included.

Reduced risk of name conflicts

Most internal symbols in libmira are no longer exported, leading to reduced risk of hard to debug name conflicts.

Change of ABI

Mira has been changed from using the VFP ABI to the ARM EABI ABI. This reduces power on nRF52 MCUs by letting the FPU remain off. See the documentation for how to use the FPU in floating point heavy applications.

Python 3

All shipped python code has been updated to Python 3 as Python 2.7 reached end of life and was deprecated on January 1, 2020.

Improved debugging capabilities after watchdog reset

The program counter from previous reset is now available via the Mira API.

Introduced local recalibration

A node on the verge of losing sync with the network can now attempt to recalibrate in order to stay synced.

Bugfixes

FOTA

- Resolved an issue which may cause a FOTA process to restart while validating an already received image.
- Resolved an issue which may cause data corruption when writing to flash with SoftDevice present.

Increased stability in Border Gateway serial port

Buffer size is increased to reduce the risk of buffer overflow and subsequently lost packets. Other general stability improvements have also been implemented.

General network stability improvements

- Resolved an issue where running concurrent BLE could prevent Mira from building a network.
- Increased tolerances to unfavourable network topologies with regard to keeping the network time synced
- Reworked routing protocol algorithms to increase preference of good parent ETX in favour of few hops to root
- Resolved an issue where internal Mira messages could be prevented from being sent
- Resolved an issue where frequency hopping sequences could be prevented from being updated properly

Known issues

Time to rebuild network after a root reset has been increased

This is a consequence of changes related to increased stability. Restarts of the root node, intentional or not, are discouraged.

Limitations

MKW41Z not supported

The MKW41Z CPU is not supported in 2.4.0. Support will be reintroduced in an upcoming release.

Asymmetric link vulnerability

It has been discovered that Mira routing protocols can have unexpected behaviour when running in configurations with asymmetric radio links. This can lead to packet loss and/or instability. Running Mira in configurations with asymmetric radio links is discouraged.