Release Note

Reach Thunderboard v346, February 2024

The team has made a number of changes to the Reach stacks since the initial release last month. We always strive for backwards compatibility, but with this release we have chosen to make a number of changes that improve consistency with a possible impact on compatibility. We attempt to communicate these clearly.

Most of the changes provided here are driven by product feature requests as we have integrated Reach into a number of IoT products.
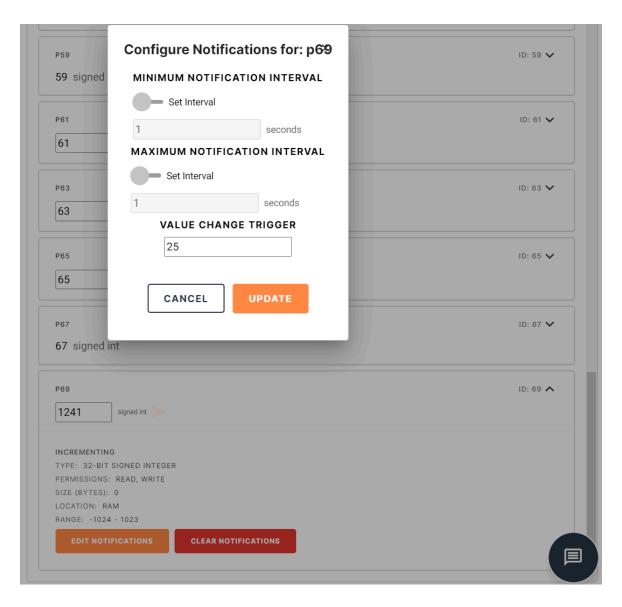
The release consists of:
- Updated protobuf specification file, version 12.
- Updated C stack, version 2.4.3
- Updated Thunderboard application, version 3.4.6
- Updated documentation
- Updated web page
- Updated mobile apps

New Features

Parameter Notifications

The latest web application supports configuring parameters for notification. As illustrated in the screenshot below, the client can request that a device notify the client on the change of a parameter. P69, the last parameter in the repository, is an integer that increments once per second. In the illustration, the dialog box requests a notification if the parameter has changed by 25 since the last notification. You can also specify limits on the interval of the notification. The minimum interval specifies that notifications will be at least this far apart. The maximum interval specifies that a notification will be generated at this rate regardless of the delta.
Note that you must have the updated Android app to use this feature. The old app crashes.

Firmware Update

The Thunderboard supports firmware update via BLE using the "EFR Connect" mobile application from Silicon Labs. The required ".gbl" file is provided with this release. You can also use the "Simplicity Commander" to load the latest hex file as described in the Getting Started document.

Time Service

The latest web application supports the "set time" and "get time" commands in the time service. This is designed to support developers setting the RTC in a device. The time is specified in a 64 bit number of seconds of the Linux epoc. An optional time zone offset encourages the system time to be kept as UTC, with a separately managed offset.

Compatibility Considerations

- As the "cr_stack.c" became larger we decided that the "one file" approach was becoming a liability. The C stack and the application have been refactored to isolate services into files. While this will make life easier for new programmers, it will look like a big change between this and the previous release. The change is more structural than logical.
- One user feedback noted the inconsistency in the naming of responses to protobuf messages. The team decided to bite the bullet and unify around xxxResponse. Hence xxxReply and xxxResult have been changed to xxxResponse.
  - As the names of these messages are not conveyed in the protobuf messages, the changed names are not noticed in deployed applications.
- The "sizes_struct" used by the client to configure communication is updated. This should have no impact on an application. The contents of the structure are unchanged so this is a binary compatible change. The change will support future growth.
- As the Silicon Labs support is not really part of the C stack, the supporting files moved to an Integrations/SiLabs directory.

Specific Changes

Issue 1: Separate stack version from app version
The Thunderboard code, like any Reach server, has three basic components. Each of these is tracked by a version number. The application version and the Reach protocol version are communicated in the device info structure. The Reach C stack now has a version which is visible in parameter ID 23 as 2.4.3.

Issue 2: Support parameter notifications
See new features, above.

Issue 3: Add Doxygen comments
The public API of Reach has been augmented with Doxygen format comments. Work is underway to support a full HTML document tree for the Reach C stack.

Issue 4: Refactor sizes structure
See the comment about sizes_struct under compatibility consideration.

Issue 5: Enable exclusion of unused services
Each service can now be included or excluded based upon #defines typically found in reach-server.h. Undefining the INCLUDE_xxx_SERVICE macro will individually exclude all reference to the given service. This is useful to reduce the memory required for code.

Issue 6: Make advertised name length variable by user
The name of the device that will be advertised via BLE may be longer or shorter based on the BLE implementation. It is 16 bytes by default, but it can be made longer using the APP_ADVERTISED_NAME_LENGTH macro in reach-server.h.

Issue 7: Add a callback when a file is completely written.
Two new crcb weak callbacks are added to support file transfers. crcb_file_transfer_complete() is called when the Reach stack has received the "is_complete" message from the server. The application can then close the file. crcb_file_prepare_to_write() is called when the stack receives the file transfer init message. The device application can use it to erase flash memory in preparation for receiving data.

Issue 8: Add time service
See above, and more notes in the programmers reference.

Issue 9: Handle the case in which there are no extended parameters.
In the previous release a device exporting no extended parameters (no enums, no bitfields) would hang up the client. This is resolved in the C stack.

Issue 10: Finer control of logging exclusion by #ifdef
The intent is that all logging can be removed by #define NO_REACH_LOGGING in reach-server.h. This is useful when there is not much memory available. There is a request to exclude log messages from the build in accordance with the log masks. This request remains open.

Android App Update: Fix crash on parameter notification
The Android app released in January would crash if a parameter notification was encountered. Parameter notifications can only be enabled in the web interface.

Reach only active when BLE is active
The Reach process function is now not called unless the BLE connection is active.

Open Issues

The web interface is "ahead" of the mobile apps. The web interface supports a number of features that are not available in the mobile apps. These include the configuration of parameter notifications and the time service.

Chrome remains the most tested usage pattern.
- Users are advised to use Chrome to access the web portal.

Restarting the device when the mobile app is connected may result in the mobile app timing out when re-reading parameters. Restart the mobile app if this happens.

Attempts to reconnect to a customer support session can result in non-functional video and screen sharing.
- Users should stop and restart the mobile app in between each support session.
- Users should end a support session and refresh their browser if they lose connection to the device.

There are some differences between the iOS and Android apps.
- The Android app has screen drawing, the iOS app currently does not.
- Reconnecting to a session on the iOS app leads to camera and screen sharing automatically restarting if they were previously shared in that session.
- A support session on iOS will go into a disconnected state within a matter of seconds when the app becomes minimized.
- The behavior of the Android app can be better or worse depending on the version of Android that is in use.
  - Accepting a request from the web agent to share your screen after having already shared your screen once in the session can crash the app
  - Canceling out of the "start casting with cygnus reach" pop-up that fires when you first go to share your screen will disable screen sharing for that session
- The iOS application has been seen to crash when the ping command is used.

Some features are not fully implemented yet.
- Session History is incomplete, in part because it expects to have a well defined product, customer and serial number. These are not necessary in the demo app.