****

**Lua Hardware Access**

**Programming Interface**

**User and Technical Manual**

Version 1.0 October 1, 2013

**Author**: Preethi Srinivasan preethi\_srinivasan@apple.com

**Introduction**

**What is LHA?**

LHA is a programming interface provided within EFI diagnostics on mobile platforms, including iPhones, iPads, iPods, and AppleTV. Its goals are to enable better lua scripting by providing more flexibility to access the hardware.

**Audience**

This document is intended for the engineers involved with the development and use of the EFI diagnostic environment. This document is written with the assumption that the users will have fundamental knowledge of Lua. For more information on Lua Fundamentals, refer to Smokey User Manual.

On the producer side, this manual addresses EFI diagnostic firmware engineers, the people who will maintain and enhance LHA.

On the consumer side, this manual addresses factory and non-factory users. Factory users include station DRIs. Non-factory users include hardware engineers needing a platform for validation or stress testing.

**Document Objective**

This document is generally intended to be an all-around reference, but written with the intent of bringing LHA users up to speed about the environment.

On the producer side, this manual intends to educate software engineers about the functionality and intent of the system. The goal is to exposit the design and shed light on the considerations required to either maintain or enhance the software.

On the consumer side, this manual describes how to use LHA. Additionally, time will be spent to document requirements, prerequisites, and expected behaviors. The goal is to arm users with the knowledge to bootstrap their own test sequences.

**LHA Fundamentals**

LHA is a programming interface provided to the user to be able to access the hardware buses- I2C, SPI, UART and access Memory and GPIO’s directly via the script. It basically exports a new library, with the name “hw” along with the libraries currently exposed by Lua- like “math”. These library functions can be accessed directly by a lua script, just the way a script would be written to call functions exported by the Lua library or it can also be run via the Smokey interface. Since they are just Smokey scripts, there is no learning required from the user as the rules applicable to run Smokey apply to this as well. It enables the user to perform bus transactions by directly calling into API’s instead of having to run commands via the “Shell” interface through the Smokey script.

**Using LHA**

The hardware access library, which goes by the name “hw” is dynamically loaded when it is required. So, the first step to ensure that you can access the LHA functions is to use this line at the top of your Smokey script-

require ‘hw’

Once this line is included, the library is loaded and all the functions exposed by it can be accessed. The LHA lib consists of a table of supported protocols, each of which points to another table which contains the functions exported by each protocol. The syntax for ‘hw’ lib calls is-

hw.<protocol name>.<function name>(arguments)

There is also a set of helper functions that can be used to perform endian conversion and byte swapping which may be required in cases like the SPI bus transmission where the endianness can be configured.

A list of supported API’s along with the syntax and data types is provided here-

<https://attache.apple.com/AttacheWeb/dl?id=ATC7a97a606690944d58d697923f23adb8e>

A sample script that runs on J72 is provided here-

<https://attache.apple.com/AttacheWeb/dl?id=ATCef9afe1bf4a342478464589198d5926a>

A keynote that describes the purpose of LHA is provided here-

<https://attache.apple.com/AttacheWeb/dl?id=ATC5c7fa7bb768e4e1793f19d5cfc1a7d1b>

This can be used as a reference to write scripts for other platforms.

Improvements are being made to LHA to support more buses like USB. This document would be updated as and when improvements are made.