Thread 1.0.1.1

Silicon Laboratories, Inc.

September 4, 2015

Note **THREAD_HOME** refers to the location where the Thread stack is installed on your machine.

1 Introduction

Thread is a secure, wireless mesh networking protocol. It is an open standard built upon a collection of existing IEEE and IETF standards, developed by Silicon Labs in conjunction with the Thread Group (http://www.threadgroup.org/).

The Thread stack supports IPv6 addresses and provides low-cost routing to other IP networks. It is optimized for low-power and battery-packed operations, and also wireless device-to-device communication. It is specifically designed for Connected Home applications where IP-based networking is desired.

Please see **THREAD_HOME**/documentation/UG10311-AppDevFundamentals-Thread.pdf for detailed information about the stack.

2 GA release

2.1 Prerequisites

- Thread stack version 1.0.1.1

- Ember Desktop version 3.3.1915
- ISA3 Utilities version 4.5.1.0

2.2 Package contents

The Thread release package contains the following directories under **THREAD_HOME**:

- app/
  Contains application framework files and application utilities (such as serial and command interpreter utilities) that aid in developing a sample application that uses the Thread stack.
- build/
  Contains the following libraries:
  - HAL library
  - Manufacturing library
  - Simulated EEPROM library
  - Thread stack library, containing stack functionality
- documentation/
  Data sheets, application development fundamentals, and other important reading material.
- hal/
  Contains board headers and other include headers pertaining to the HAL library.
- ncp-images/
  Contains binary images for use on a network coprocessor (NCP).
- stack/
  Contains stack include headers that are necessary for building a sample application.
- tool/
  Contains bootloaders.

2.3 Installation

These instructions assume that you have done the following:

- Opened your EM35x Development Kit.
- Browsed the link to the Quick Start Guide.
- Followed the instructions to register your development kit.
- Downloaded this version of the Silicon Labs Thread stack.

2.3.1 Software System Requirements

- Microsoft Windows XP or later
- Java version 1.6 or later. The Ember Desktop development environment must be used with a 32-bit JRE. If you are using a 64-bit system, you must install a 32-bit JRE.
- Adobe Acrobat Reader (available free from http://get.adobe.com/reader/)

2.3.2 About Installing the Silicon Labs Thread Stack

Always install the stack first. When you run the installer executable, it performs the following actions:

- Installs the files needed to develop applications on the stack into a directory specified by the user.
- Adds a shortcut to the Start Menu into the folder specified by the user.
- Adds an entry to "add/remove programs" (accessed from Windows Control Panel) so the stack can be uninstalled.

This does not affect any environment variables. If you have previously installed the stack, install this release in a different folder. Multiple installations can exist together.

2.3.3 Installing Ember Desktop

Install Ember Desktop by downloading the self-extracting installer from the support portal and running it. If you do not have 32-bit JRE installed, the installer and the application will report an error on startup.

2.3.4 Installing IAR Workbench for ARM

IAR Embedded Workbench for ARM (IAR-EWARM) is not downloaded from the support portal. An IAR Embedded Workbench Download Card is included with your EM35x development kit. Depending on which development kit you ordered (IAR or not), you will receive either a 30-day evaluation version of EWARM or a licensed version. If you received a full development kit (EM35X-DEV-IAR), you will also receive a Welcome Letter with full development license information. To install the IAR Embedded Workbench for ARM:

- Download the installer from the link provided in the license information. The executable will extract files to a temporary folder and then launch the installer. Click on the "Install IAR EWARM Embedded Workbench" link and follow the install instructions.
- Refer to the "Installation and licensing information" section of the IAR installer for additional information about the installation process and how to configure your license.

If you are receiving this release as a product upgrade, the installer for IAR EWARM 7.30.1 is available at ftp://files.iar.com/pub/silabs/EWARM-7301.exe.

2.3.5 Installing the ISA3 Utilities

Always run the ISA3 Utilities installer after all previous installers, including after IAR-EWARM. Download the installer from the support portal and run it. The installer does the following:

- Installs the ISA3 drivers so that the debug adapter (ISA3) can be used both over USB and Ethernet.

- Installs the following command line utilities:
  - em3xx_buildimage.exe
  - em3xx_load.exe
  - em3xx_isa.exe
  - em3xx_convert.exe
  - stm32_load.exe
- Adds an entry to "add/remove programs" (accessed from Windows Control Panel) so that the utilities can be uninstalled.
- Modifies your PATH environment variable so that the command line utilities can be easily executed from a Windows Command Prompt.

2.3.6 Installing FTDI USB Drivers

To use the FTDI USB interface of the Breakout Board for UART connectivity, you must install the drivers for the FTDI USB-Serial converter. You can obtain this driver from FTDI at http://www.ftdichip.com/Drivers/VCP.htm. These drivers are not required if you connect to the UART of the Breakout Board using the RS-232 port or via the pass-through UART functionality of the Debug Adapter (ISA3).

2.3.7 Installing Silicon Labs USB Drivers

To use the EM358x USB interface of the Breakout Board for UART connectivity, you must install the drivers for the Silicon Labs USB-Serial converter. This driver is included in the Ember Desktop installation as an optional component, but you can obtain this driver from Silicon Labs at http://www.silabs.com/.

2.4 Stack API documentation

The Thread stack API reference guide is Doxygen-generated documentation that is rooted at **THREAD_HOME**/documentation/Thread-Doxygen/index.html.

This document contains a comprehensive list of APIs used to interface to the Thread network. These APIs concern network management, device and stack management (including management of stack tables, event scheduling, message buffers and security), messaging, fragmentation, serial communication, token access, peripheral access, bootload utilities, etc.

2.5 Ember Desktop

The Ember Desktop package includes both network analyzer and AppBuilder support.

- The Network Analyzer is a graphical tool that displays network and node activity in real time. It also provides an interface for loading new applications through the debug adapter.
- The AppBuilder is a graphical tool used to create configuration and build files for the Application Framework. The configuration files created by AppBuilder indicate which components and configuration are built for the compiled binary image.

Please see included documentation in the Ember Desktop package for more information.

3 Known issues

- This release supports EM358x and EM359x SoCs and NCPs.
- The Thread specification and certification process is incomplete so the stack and related APIs may change without notice.
- For this release, we are including a preliminary application layer that will be fleshed out further over time.
- Long running networks with a mix of sleepy end devices and always-on devices can sometimes detach from the network.
- There is currently no support for an NCP over SPI. This will come in a future release. If this is important to you, please communicate this to customer support so we can give this the proper priority.

4 Fixed issues

- An include file problem, which prevented host applications from compiling, has been resolved.
- A bug has been fixed that caused multicast messages to silently fail when attempting to send from a host to the mesh.

5 Building a sample application with Thread support

Following is a handy guide to creating a sample application using AppBuilder.
Additional information is available at **THREAD_HOME**/app/thread/doc/README.

5.1 Load the Thread stack in AppBuilder

1. Launch Ember Desktop.
2. Choose File > Preferences from the menu.
3. Select Network Applications from the list on the left.
4. Click Add... on the right.
5. Browse to the location where you installed the release.
6. Click Open.
7. Verify the list of installed stacks now contains Thread.
8. Click OK.

5.2 Create a new application based on a sample

1. Choose File > New > Application Framework Configuration... from the menu.
2. Select Thread from the list of application types.

3. Click Next.
4. Select the Thread stack you loaded in the first part.
5. Click Next.
6. Select Start from a selected sample application.
7. Select one of the provide sample applications.
8. Click Next.
9. Specify a location for your application.
   Note: the location must be on the same Windows partition as **THREAD_HOME**.
10. Specify a name for your application.
11. Click Finish.

5.3 Generate and build your new application

1. Click Generate.
   Note the location of generated files, particularly the IAR Embedded Workbench workspace file (.eww) or Makefile.
2. Click OK.
3. Build the application:

   SoC Open the workspace file (.eww) in IAR Embedded Workbench and choose Project > Make.

   Host Run `make` from your application directory.