MSP430 Release Notes
Release Notes
Release v1.5 R2 (4.0.1.10)

Stonestreet One
9960 Corporate Campus Drive
Suite 3300
Louisville, KY 40223

Copyright 2014 Stonestreet One, LLC All rights reserved.
This document is the proprietary property of Stonestreet One, Inc., Louisville, Kentucky, USA and is considered CONFIDENTIAL.

Revision History

Rev	Updated Areas	Date	Author
0.1B	First Release	02/27/12	T. Cook
0.1	Updated to CCS5	03/28/12	T. Cook
0.2	Added LE Support.	04/20/12	T. Cook
0.5	Added LE sample applications	06/04/12	D. Lange
0.9	Added SPP+LE Combined Application	07/06/12	T. Cook
1.0	Bug fixes + Patch Update	07/24/12	T. Cook
1.1	Bug fixes + SPPDemo added	07/30/12	T. Cook
1.2	Comprehensive Release Update	10/10/12	T. Cook
1.3	Comprehensive Release Update	07/08/13	T. Cook
1.4	Audio update.	07/25/13	T. Cook
1.4 R1	Audo Sink Ref. Board Support	11/26/13	T. Cook
1.4 R2	Audio Source Ref. Board Support	4/24/14	T. Cook
1.5	Multi-source and multi-room audio SNK demos, PXP/FMP demos	6/4/14	M. Brown
1.5 R1	Updated patches from v1.5	7/18/14	T. Cook
1.5 R2	Reduced demo code size	9/10/14	M. Brown

Table of Contents

1. Changed in Release 1.5 R2 (Bluetopia 4.0.1.10)

This release contains changes to the user HAL code (no library changes) to reduce the total code size of the demos. Comprehensive changes are listed below:

1.1 Eliminated HAL floating-point dependency

The dependency on floating-point calculations for calculating a UART baud rate has been removed from HAL.c for all platforms. For unmodified demos this has the net effect of reducing code size by about 1.5 KB.

2. Changed in Release 1.5 R1 (Bluetopia 4.0.1.10)

This release contains support for updated Audio related profiles and additional LE profiles. Comprehensive changes are listed below.

2.1 Updated patches

This release contains updated patches as provided by TI.

3. Changed in Release 1.5 (Bluetopia 4.0.1.9)

This release contains support for updated Audio related profiles and additional LE profiles. Comprehensive changes are listed below.

3.1 Updated Core Library

This update contains the latest version of the Bluetopia core library. In addition to some minor bug fixes, this library contains an updated GAVD and AUD API as detailed below.

3.2 Updated AUD library API

The existing A3DPDemo_SNK and A3DPDemo_SRC demos have been updated to use the new AUD API, which allows for multiple connections to a single registered AUD SNK or SRC, specified by BD_ADDR. A3DPDemo_SRC still only supports connecting to a single remote sink; however A3DPDemo_SNK has been updated for multiple incoming sources (detailed below).

3.3 Multi-Source support for A3DPDemo_SNK

A3DPDemo_SNK has been modified to allow up to two incoming A2DP sources. When a new source connects in, or an existing source attempts to start streaming audio, any currently-playing streams will be paused.

3.4 New A3DPMultiRoomDemo

A modified version of A3DPDemo_SNK has been created to make use of new experimental A3DP features in which audio from an incoming source can be forwarded to a second speaker, allowing for multi-room setups with a single connection from a phone or other A2DP source. This new demo requires the use of two receivers running the demo, as well as debug UART access on both to set up the connection.

The files for this new demo can be found in the MSP430_Experimentor\Samples\A3DPMultiRoomDemo directory.

3.5 New Low-Energy Profile Demos (PXPDemo, FMPDemo)

Demos have been added to support the following Bluetooth Low-Energy profiles:

- Proximity Profile (PXP)
- Find Me Profile (FMP)

These demos can be found under the PXPDemo and FMPDemo directories inside the standard sample directory of MSP430_Experimentor\Samples\, for both IAR and CCS configurations.

As a result of adding these profiles, the following LE services have been provided in both library and source code form:

- Link Loss Service (LLS)
- Immediate Alert Service (IAS)
- Transmit Power Service (TPS)

4. Changed in Release 1.4 R2 (Bluetopia v4.0.1.8)

This release contains support for Audio related profiles. Comprehensive change list is below.

4.1 Audio Sink Reference Board Support

Added support for TI's Audio Source Reference board based on a MSP430F5229. This support is added to the IAR project for the A3DPDemo_SRC sample application. Modified files in this release are:

- MSP430_Experimentor\
  - Hardware\A3DP_SRC — Directory added
  - Samples\A3DPDemo_SRC
    - A3DPDemo_SRC.c — Added
    - Main.c — Added
    - Main.h — Added
    - ti_cap_lib\ — Added directory
    - Projects\IAR — Added IAR project files

5. Changed in Release 1.4 R1 (Bluetopia v4.0.1.8)

This release contains support for Audio related profiles. Comprehensive change list is below.

5.1 Audio Sink Reference Board Support

Added support for TI's Audio Sink Reference board based on a MSP430F5229. This support is added to the IAR project for the A3DPDemo_SNK sample application. Modified files in this release are:

- MSP430_Experimentor\
  - Bluetopia\
    - btvs\BTVS.c Changed
    - include\BTPSCFG.h Changed
    - profiles\A3DP\lib\IAR Updated IAR A3DP Libraries.
  - Hardware\A3DP_SNK Directory added
  - Samples\A3DPDemo_SNK
    - A3DPDemo_SNK.c Changed
    - Main.c Changed
    - Main.h Changed
    - ti_cap_lib\ Added directory
    - Projects\IAR Modified IAR project files

5.2 Changed revision numbers in Release Document

The revision numbers in this release notes to match the release numbers used by TI for the SDK. The Bluetopia stack version is noted for each release number

6. Changed in Release 1.4 (Bluetopia v4.0.1.8)

This release contains support for Audio related profiles. Comprehensive change list is below.

6.1 Core library changes

This release contains changes a new set of libraries with higher MTUs that are needed for the A3DP functionality. There are now two new folders for each compiler: DefaultMTU and LargeMTU. The DefaultMTU contains core libraries built with a L2CAP MTU of 335 bytes. The LargeMTU contains core libraries built with a L2CAP MTU of 800 bytes that is needed for A3DP.

6.2 HFP 1.6 Support Added

This release contains support for Hands Free Profile version 1.6 (Hands Free role only) with Wide Band Speech (WBS) support. There is an accompanying demo, HFPDemo, which provides a command line interface to demonstrate this functionality.

6.3 Assisted A2DP Support Added

This release contains support for assisted A3DP (Sink Role only). There is an accompany demo, A3DPDemo_SNK, which demonstrates this functionality.

6.4 AVRCP Support Added

This release contains support for AVRCP (Target and Controller roles). There is an accompany demo, A3DPDemo_SNK, which demonstrates the AVRCP Controller role.

7. Changed in Release 1.3 (Bluetopia v4.0.1.7)

This release contains many comprehensive changes to the release. Support has been added to all of the projects for the CC256xB chipset. Support has been added for the HID over GATT Profile using LE and a sample application, HOGPKeyboardDemo, has been added that demonstrates this profile. All of the updates are described below.

7.1 New Core Library

This release contains the latest Bluetopia Core stack. Reductions to the code size that are used by the stack have been made as well as changes to allow the linker to automatically strip some code segments that are not used by a particular application.

7.2 Larger L2CAP MTU

This release contains support for L2CAP packets up to 339 bytes in length. This will significantly improve the maximum achievable throughput that is possible.

7.3 UUID MACRO Changes

All UUID MACROs that were contained in BTBTypes.h now have either BLUETOOTH or SDP in the name to denote if they take (or assign) either SDP UUIDs (which are big endian) or Bluetooth UUIDs (which are little endian). The Bluetooth UUID MACROs take the UUID in Big-Endian form and convert to little-endian as needed.

7.4 GDIS Support removed

GATT Discovery Module (GDIS) has been removed in this release. Equivalent functionality is now built directly into GATT library.

7.5 HCILL moved into Core Stack

In this release the HCILL implementation is contained in the Core stack. HCILL.c and HCILL.h are no longer shipped. Now the user must select the HCILL protocol in the HCI driver information that is passed to BSC_Initialize() in order to enable HCILL. All the sample applications do this by default.

7.6 GATT Packet Queueing

The GATT library contains a new feature that allows a mechanism of limiting the maximum number un-acknowledged transactions (Write Commands and Notifications) that can be outstanding at a time per LE connection. SPPLEDemo and SPPLEDemo_Lite sample applications use this feature to limit the RAM consumption when the remote device grants a large number of credits.

7.7 HID over GATT Profile (HOGP) Added

Support for the HID over GATT profile (HOGP) has been added in this release. A new sample application, HOGPKeyboardDemo that demonstrates the HID device role has also been added.

7.8 Battery Service (BAS) Added

Support for the Battery Service was added as a requirement for supporting HID over GATT (HOGP).

7.9 BTPSVEND Changes

7.9.1 Patch Header Naming

The patch header naming convention has been changed in this release to no longer include Panasonic module name in the header file name. The new naming convention is "CC256X".

7.9.2 CC2560 Support removed

This release no longer ships a patch header for the CC2560 chipset.

7.9.3 CC256xB Support added

This release contains a patch header, "CC256XB.h", for the CC256xB chipset family. The service pack that is shipped with the release is SP 0.1.

7.9.4 Changing Bluetooth Baud Rate

BTPSVEND now contains support for configuring the baud rate that is passed as a part of the HCI driver information that is passed as a parameter to BSC_Initialize(). If the baud rate that is passed is not equal to 115200 baud (which is the TI chip's default baud rate) BTPSVEND will automatically update the baud rate after an HCI Reset is performed to the value passed to BSC_Initialize(). This reduces the complexity of changing the baud rate in the sample applications (which all still default to 115200 baud).

7.9.5 HCILL Changes

As part of the HCILL changes detailed above BTPSVEND will now automatically enable HCILL if either the "cpHCILL" or "cpHCILL_RTS_CTS" protocols are set in the HCI driver information that is passed to BSC_Initialize(). The protocol parameters are set to some defaults but can be changed by defining some pre-processor symbols.

7.10 BTVS Changes

The BTVS source that is shipped in this release contains several new APIs. A new API was added to change the public BD_ADDR that is used by the controller. Two new APIs for configuring and enabling HCILL (called by BTPSVEND) were also added.

7.11 Accepting Connection Parameter Updates

All LE sample applications that support the LE Master role will now automatically accept any connection parameter update request that is sent by a connected slave device.

7.12 SPPLE Fixes

Several bugs that effected the buffer management, in certain cases, have been identified and fixed in this release. The SPPLEDemo and SPPLEDemo_Lite applications contain the fixes.

7.13 Multiple Connection SPPLE

SPPLE Demo now supports multiple SPP connections as well as multiple SPPLE connections (in LE master role only).

7.14 KeyFobDemo Application Changes

Several TI requested changes were made to KeyFobDemo. In particular the format of the ASCII data that is sent in ASCII mode was changed at TI's request. ASCII mode is also the default mode out of the box.

8. Changed in Release 1.2 (Bluetopia v4.0.1.6)

This release contains many comprehensive changes to the release. Support has been added to all of the projects for the ez430-RF256x board, and support has been added the new sample application SPPLEDemo_Lite for the MSP430F5529 Experimenter Board. Several new sample applications have been provided as well as a new BR/EDR profile (HID). A new mechanism, called the Flexible Build Library (FBL), that allows customers to create versions of the provided libraries to match their specific use case has also been provided. All of the updates are described below.

8.1 New Core Library

This release contains a version of the Core library that is compiled with LE support (libBluetopia_LE.a) and a version that is compiled without LE support (libBluetopia.a). A version of the core library built using the Flexible Build Library (FBL) is also provided as well. See the "Provided Libraries.pdf" document for more explanation on the new libraries that have been provided.

8.2 Patch RAM Update

This release contains the Service Pack 2.5 base patch and LE patch.

8.3 Vendor Specific Helper Module

The vendor specific command helper source code has been included under "Bluetopia\btvs". This is the same source provided for the Stellaris release with updates to correctly set the LE transmit power level.

8.4 HID (BR/EDR) Support

BR/EDR support has been added in this release. The headers and the HID library is provided under "Bluetopia\profiles\HID" that applications can link against to add HID functionality to their applications. A new sample application, HIDDemo, has been provided that shows how to perform both roles in HID (Host and Device).

8.5 ez430-RF256x Support

The ez430-RF256x device is supported in this release. In the directory "Hardware\ez430" are the platform files that have been added to support this hardware platform. All of the provided sample applications now have selectable build options that can be selected to run on this hardware platform.

8.6 MSP430F5529 Experimenter Board Support

Support for the MSP430F5529 Experimenter Board has been added to the SPPLEDemo_Lite for IAR. In the directory "Hardware\MSP430_EXP5529" are the platform files that have been added to support this hardware platform.

8.7 SPPLEDemo_Lite added

A new sample application, called SPPLEDemo_Lite, has been added. This application demonstrates performing the SPP Server role, the LE Slave role, and the GATT Server role. This application allows another device to connect over BR/EDR or LE and send data (using SPP for BR/EDR, using SPPLE proprietary server for LE). Any data received by a device running this application will then be looped back to the sender.

8.8 KeyFobDemo added

A new sample application, called KeyFobDemo, which emulates the CC2540 Key Fob has been added. This device can be connected to over LE or SPP and will send accelerometer and button press data to any connected device. As this emulates the Key Fob any device running this application can be connected to using the application (TI-BLE-Demo) that TI provides as source code for iOS.

8.9 Flexible Build Library (FBL)

This release contains support for what is being called the Flexible Build Library and is a wonderful enhancement to the release. All of the FBL related files can be found under "Bluetopia\Objects". The FBL allows the customer to generate a version of the core library matched to their application needs, and versions of the GATT and HID libraries tailed to their application's role requirements. The features described below can be either supported or not supported as need by the customer. Any option that is not specified on the command line of the FBL script (fbl_ccs.pl or fbl_iar.pl) is not added to the libraries generated by the script.

- SPP Server Specified by "--sppserver".
- SPP Client Specified by "--sppclient".
- SDP Client Specified by "--sdpclient".
- LE Master Specified by "--lemaster".
- LE Slave Specified by "--leslave".
- SCO Audio Specified by "--sco".
- GATT Server Specified by "--gattserver".
- GATT Client Specified by "--gattclient".
- HID Host Specified by "--hidhost".
- HID Device Specified by "--hiddevice".

This allows the customer to match the library to their use case and not have the penalty of supporting the other use cases. For example a customer who needs a library with SPP client, and SDP client support can specify the following (which generates a new libBluetopia.a for IAR):

fbl_iar.pl --sppclient --sdpclient

A customer who needs HID device, GATT server and LE slave support can specify the following (which generates a new libBluetopia_LE.a, new libSS1BTHID.a, and new libSS1BTGAT.a that the customer can link with):

fbl_iar.pl --hiddevice --leslave –gattserver

9. Changed in Release 1.1 (Bluetopia v4.0.1.5)

This contains contain some bug fixes, as well as contains the SPPDemo application (which is BR/EDR only).

9.1 Bug fixes

Fixed bug in SPPLEDemo where “Pair” command would be executed when “PairLE” was typed. Also fixed bug where “Quit” command did not work properly.

9.2 SPPDemo now included

Added BR/EDR only sample application back into release.

10. Changed in Release 1.0 (Bluetopia v4.0.1.4)

This contains various bug fixes, the biggest being memory allocation errors. The Patch RAM has also been updated to the latest version provided by TI.

10.1 Bug fixes

Fixed various bugs in the applications.

10.2 Memory Allocation Fixes

Ported changes memory allocator to use less memory for the headers. Also fixed issues where applications ran out of memory due to fact that projects did not allocate enough memory in the preprocessor settings.

10.3 Patch RAM Update

Updated the base service pack and the Low Energy add on pack to Service Pack v2.4.

11. Changed in Release 0.9 (Bluetopia v4.0.1.3)

This release contains an application that performs LE and SPP functionality in 1 sample application (SPPLEDemo). This also includes a reduced RAM footprint (under 7K requirement with application).

11.1 SPP + LE Sample Application

Added SPP support to existing SPPLEDemo. This demonstrates using SPP and also using LE in 1 sample application.

11.2 Reduced RAM Requirements

Reduced RAM footprint (under 7K requirement with application).

12. Changed in Release 0.5 (Bluetopia v4.0.1.2)

This release contains increased Low Energy support. This support includes profiles/services and sample applications for:

- Alert Notification Profile (ANP)
- Heart Rate Monitor Profile (HRP)
- Health Thermometer Profile (HTP)
- Phone Alert Status Profile (PASP)

12.1 Increased Low Energy Profile Support

Added increased Low Energy support, as well as a sample application that uses the Generic Attribute Profile and code implementing the above mentioned profiles.

13. Changed in Release 0.2 (Bluetopia v4.0.1.1)

This release contains Low Energy support, as well as IAR support.

13.1 Low Energy Support

Added Low Energy support, as well as a sample application that uses the Generic Attribute Profile.

13.2 IAR Support

Added support for IAR.

14. Changed in Release 0.1 (Bluetopia v4.0.1.1 Beta)

This release is the first official release.

14.1 Updated to CCS5

Updated project to use CCS v5.

14.2 Low Power Scan

Add Low Power Scan configuration to patch download.

15. Files Included in this Release

15.1 Documentation

Various Standard Documents are provided in PDF form. These documents include developers guide and API documents for the core Bluetooth Stack.

This information is provided in the Documentation folder.

15.2 Source and Libraries

All end user source and libraries are provided in a Zip file format. The archive name is ‘TI_Bluetopia_4_0_1_10_MSP430.zip’.

16. Release Notes

16.1 Build Notes

This release covers a Bluetopia Release that supports Bluetooth 4.0. This release is intended to be built with either Code Composer Studios for MSP430 or IAR's Embedded Workbench for MSP430. Projects for both are included with the correct include path and other settings to be built out of the box with little effort.

16.2 O/S Abstraction Notes

The O/S abstraction layer is specified in the BTPSKRNL.c/h file.

16.3 Transport Notes

The HCI Transport abstraction layer is specified in the HCITRANS.c and HCITRANS.h files.

EHCILL.c/.h, as well as HCITRANS.c, supports the TI specific EHCILL protocol for power management. In order to enable this protocol the HCILL_Configure (defined in EHCILL.c/.h) function should be called with the desired protocol parameters.

A vendor specific file, BTPSVEND.c, is included that downloads the Patch RAM to the CC2564.

This version supports the UART transport protocol. The protocol is selected by correctly populating the correct protocol in the HCI_DriverInformation_t structure. The following code snippit selects the HCILL low power protocol:

```
HCI_DriverInformation_t DriverInformation;

HCI_DRIVER_SET_COMM_INFORMATION(&DriverInformation, 1, 115200,
cpHCILL_RTS_CTS);

BSC_Initialize(&DriverInformation, 0);
```

Note that in both cases, the driver information is passed to the HCITR_COMOpen() function so that the driver can take any required action based on the parameters that are specified for the open. Also note that the baud rate can be changed here to something other than the default of 115200 baud if needed (for example to increase data throughput).