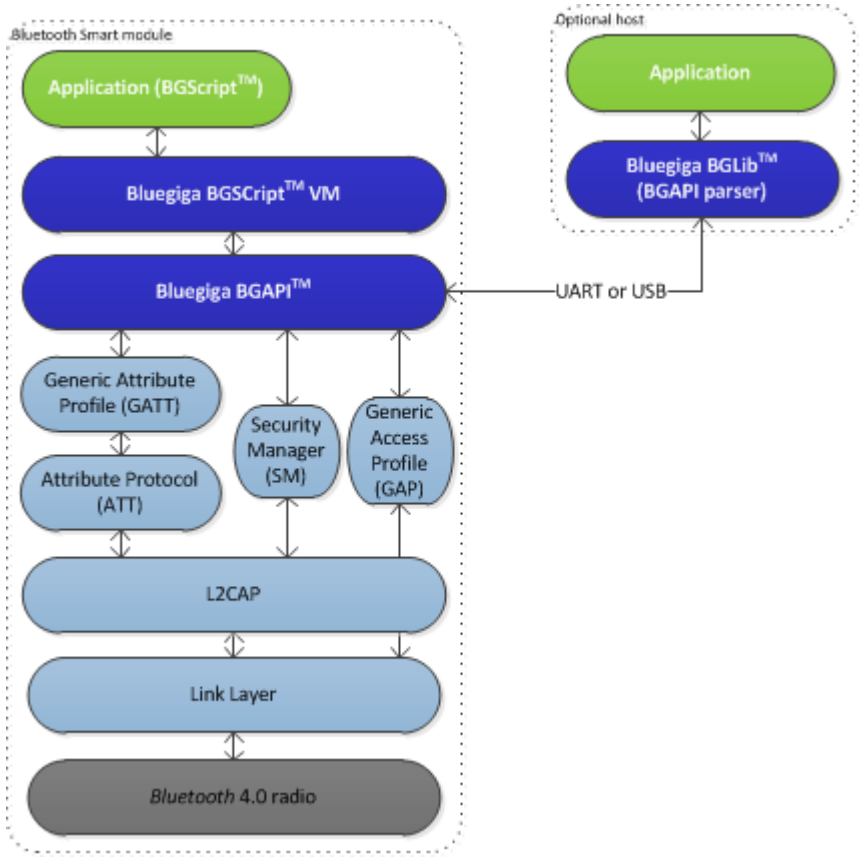
Bluegiga *Bluetooth*® Smart Software v.1.3

Table of Contents

Introduction to *Bluetooth* Smart Software

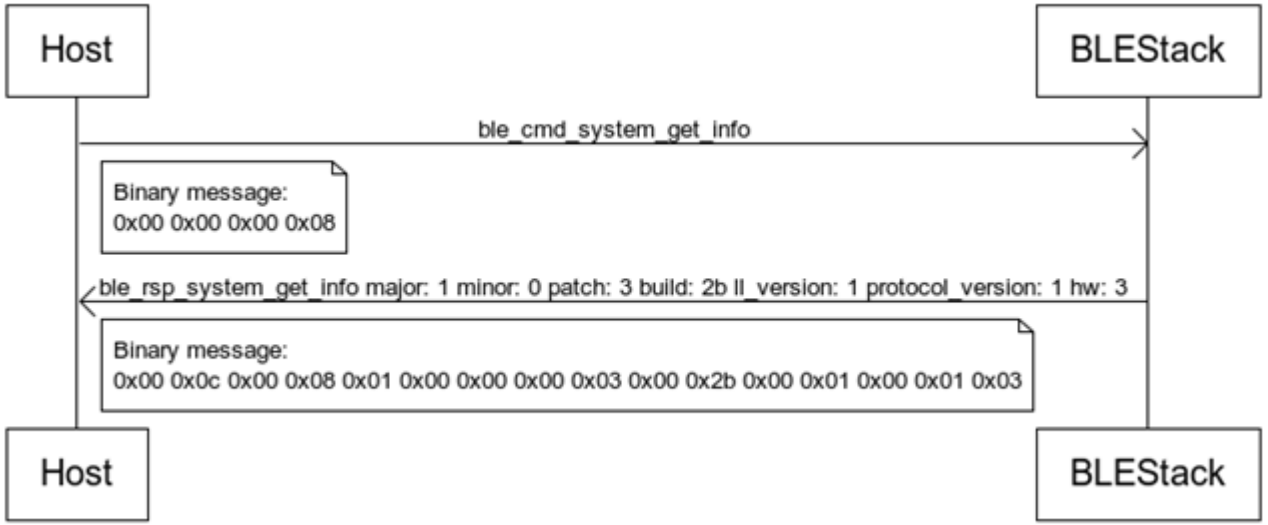
- ***Bluetooth v.4.0*, single mode compliant**
  - Supports master and slave modes
  - Up to 8 simultaneous connections
  - 100 kbps peak throughput
- **Implements all *Bluetooth* Smart functionality**
  - GAP, L2CAP, ATT, GATT
  - Security manager: bonding, encryption
  - *Bluetooth* Smart profiles
- **Simple API for external host processors**
  - BGAPI™ : A simple protocol over UART or USB interfaces
  - BGLib™ : A C library for host processors implementing BGAPI
- **Supports standalone applications as well**
  - BGScript™ : A simple scripting language for writing applications
  - Native C applications developed with IAR Embedded Workbench
  - **No separate host needed**
- **Over-the-Air firmware upgrade**
  - Stack updates
  - Application and GATT updates
- **Blutoooth Smart Profile Toolkit™**
  - XML based development tool for Bluetooth Smat profiles
  - Fast and simple profile development
- **Small memory requirements**
  - ~4-6kB RAM
  - ~80-100kB flash (depending of used features/profiles)
- ***Bluetooth* qualified**



Bluegiga *Bluetooth*® Smart Software

Introduction to *Bluetooth* Smart Software

- **BGAPI™ protocol** : A simple binary command, response and event protocol between the host and the stack
  - Used when a separate host (MCU) is used to control *Bluetooth* stack over UART/USB
  - Very small memory requirements size requirement and low implementation overhead



Introduction to *Bluetooth* Smart Software

- **BGLib™ library** : A portable ANSI C library, which implements the BGAPI protocol
  - Easy to port to various architectures such as : ARM Cortex, PIC16/32 etc, Coldfire+ etc.
  - Uses fuction–call back architecture

C Functions

```
/* Function */
void ble_cmd_gap_connect_direct(
    bd_addr address ,
    uint8 addr_type ,
    uint16 conn_interval_min ,
    uint16 conn_interval_max ,
    uint16 timeout
);

/* Callback */
void ble_rsp_gap_connect_direct(
    uint16 result ,
    uint8 conn
);
```

Introduction to *Bluetooth* Smart Software

- **BGScript™ scripting language** : A very simple BASIC-like application scripting language
  - Used when applications are implemented on the Bluetooth radios MCU
  - Enables very fast application development and allows programs to be executed directly on the *Bluetooth* radio without the need of an external MCU

```
# System boot event listener : Executed when BLE112 is started
event system_boot(major ,minor ,patch ,build ,ll_version ,protocol_version ,hw )


    # Configure ADV interval to 1000ms and start advertisements an all channels
    call gap_set_adv_parameters(1600, 1600, 7)

    # Start generic advertisement and enable connections
    call gap_set_mode(2,2)

   #Start a continuous software timer, which generates interrupts every 1000ms
   call hardware_set_soft_timer(32768, 1, 0)
end
```

Introduction to *Bluetooth* Smart Software

- **Why to use BGScript™?**

- **Very simple to use**
  - Fast development of simple *Bluetooth* Smart applications
  - Examples: Pairing, simple user interfaces, simple sensors

- **Free software development tools**
  - Code developed with any text or source code editor
  - Code compiled with Bluegiga's free compiler

- **Several example scripts available**
  - Heart Rate sensor
  - Proximity reporter
  - FindMe tag
  - Medical devices such as blood glucose

- **Cuts out the need for external MCU**
  - Reduced product eBoM
  - Smaller footprint
  - Faster time-to-market

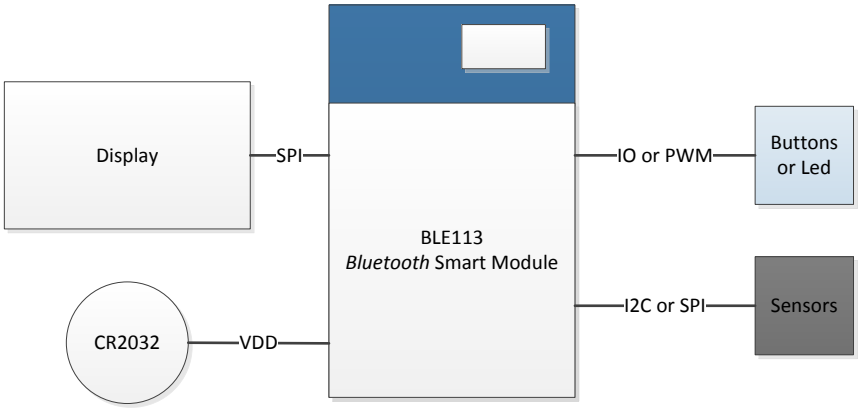
Introduction to *Bluetooth* Smart Software

- ***Bluetooth* Smart Profile Toolkit™**: A tool for creating *Bluetooth* Smart profiles
  - *Bluetooth* Smart profiles are very simple
  - Can be describes with a single file of XML
  - Profile toolkit is a Simple description language of *Bluetooth* Smart Profiles
- **Several example profiles and services available**
  - Heart Rate Sensor
  - Proximity Reporter
  - FindMe
  - Blood Glucose
  - Heath Thermometer
  - Battery Service
  - Vendor Specific services

```
<service uuid="1800">
  <description>Generic Access Profile</description>

  <characteristic uuid="2a00">
    <properties read="true" const="true" />
    <value>BG Demo</value>
  </characteristic>

  <characteristic uuid="2a01">
    <properties read="true" const="true" />
    <value type="hex">4142</value>
  </characteristic>

</service>
```

Introduction to *Bluetooth* Smart Software

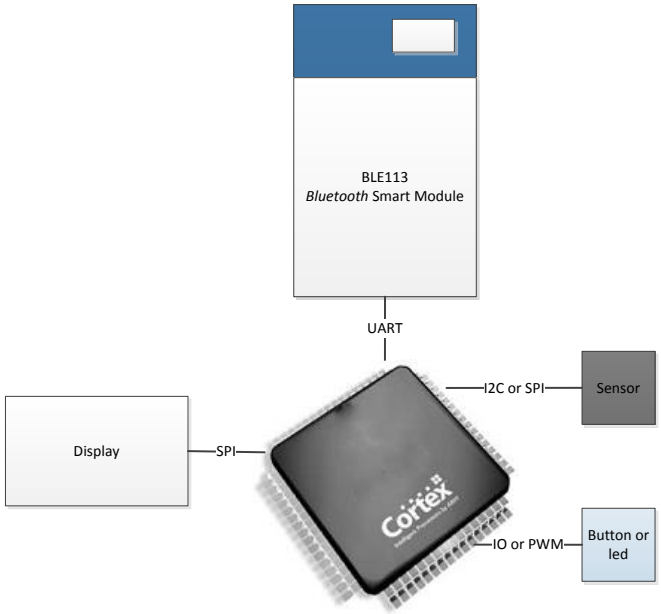
- **Standalone architecture example using Bluegiga BLE113 module**
  - Sensors and peripherals are directly connected to the BLE113 via the IO interfaces
  - Application executed on the on-board 8051
  - Application developed with BGScript™ or C SDK and services and profiles with Profile Toolkit™



Applications: sport and fitness, medical and health care, smart energy, home automation, security, proximity and precence etc.

Introduction to *Bluetooth* Smart Software

- **Hosted architecture example using Bluegiga BLE113 module**
  - Sensors and peripherals are directly connected to the MCU via the IO interfaces
  - BLE113 connected to the MCU via UART or USB
  - Application developerd to the MCU and interfacing to BLE113 done using BGAPI™ protocol (BGLib™ can be used on the host)
  - Profile developed with Profile Toolkit™



Bluetooth® Smart Software v.1.3

Feature Highlights

Highlight Features

- **BLE121LR support**
  - Added support for the new long range product variant
- **BLE113-A-M256K support**
  - Added support for the BLE113-A with on-board 256kB flash
- **Enhanced Broadcasting (Patent Pending)**
  - Allows an advertiser device to know who has received the broadcast packet
  - Allows for example better power saving, when advertisement can be paused after data has been received
  - Does NOT however guarantee the data gets reliably transmitted
- **Slave mode : Simultaneous Advertisement or Scanning While Connected**
  - Enabled advertisement or scanning when connected
- **Security : Fixed Pass Key Support for Bonding**
  - The use of a fixed 6 digit PIN code for bonding
  - This is little bit in the grey area as the standard uses random pass key

Feature Highlights

Device Firmware Upgrade (DFU):

- **Over-the-Air Firmware Upgrade**
  - Enables *Bluetooth* Stack, GATT and BGScript application to be updated over a Bluetooth Smart connection
  - v.1.3 add the possibility to update just the GATT and Application
    - OTA update ~90-95% smaller firmware update file
  - OTA support can be integrated into any customer application
  - Requires 256kB flash – either external SPI flash or on-board 256kB

- **DFU over UART**
  - Field firmware upgrades over UART

- **DFU Support Added to BLEGUI**
  - Easy firmware upgrades using BLEGUI

Feature Highlights

API Improvements

- **RF**
  - API to control the receiver sensitivity added
- **PHY APIs**
  - Scan all RF channels and report RSSI
  - During a connection measure RSSI or packet loss per channel and report the data
  - Block or enable some of the RF channels
- **Flash APIs**
  - Read, Write and Erase data from on-board flash (user data area)
- **I/O APIs**
  - Enable / disable I/O interrupts
  - Set I/O directions
- **Analog comparator**
  - API support added
- **USB**
  - API to enable/disable USB interface added (BLE112 only)
- **AES engine**
  - API to access the AES hardware engine added

Feature Highlights

BGScript™ Improvements

- **Function Support**
  - BGScript programs can now use functions (procedures)
    - Reuse of code within an application
    - More compact code
    - Simpler applications

- **BGScript code can be split into multiple files**
  - Easier code maintenance
  - Easier code updating

- **Memory management functions**
  - Memset(), memcpy() etc.

Feature Highlights

- **Examples**
  - iOS and Android App source code
  - 20+ BGSCript examples
    - Cable replacement, iBeacons, Health Thermometer, OTA update, Blood glucose sensor, HR transmitter, AT commands etc.
  - BGAPI source code for host
    - ANSI C
    - 3rd party implementations: Java, C#, Python and Arduino

- **Documentation**
  - API documentation: BGAPI, BGScript, BGLib and C SDK
  - User Guides: BGScripting, Hardware configuration and GATT services
  - Application notes: Building your 1st Bluetooth Smart Application, OTA Updates, Glucose sensor, HR sensor, C SDK etc.
  - Android and iOS development presentation, Bluetooth certification, Technology presentation etc.

Thank You