Understanding Bluetooth® Low Energy

Hary Radakichenane – RF Marketing Manager
Raffaele Riva – RF Application Manager
11:10  Bluetooth Low Energy Introduction
- Bluetooth Low Energy Power Consumption
- Maximize Bluetooth Low Energy Application Throughput
- Voice over Bluetooth Low Energy
- Over The Air (OTA) FW Upgrade over Bluetooth Low Energy

12:00  ST Bluetooth Low Energy Roadmap
Bluetooth Evolution

- First Adopted Spec
- Major Bug Fixes
- Adaptive Frequency Hopping and discovery
- Enhanced Data Rate
- Secure, Simple Pairing and NFC Support
- Power Control
- Alternate MAC & PHY 24Mbps, Enhanced
- Bluetooth Low Energy Specification
- 4G LTE Co-existence, Hub and end-point
- Higher throughput, slownet, LE Privacy v2.1
- LE 2M PHY, Long range, Advertising extension

- 1.0 1.1 1.2 2.0 2.1 3.0 4.0 4.1 4.2 5.0

Classic Bluetooth
Bluetooth Low Energy: Designed for Success

- Lower power than Bluetooth “Classic” (or Bluetooth 3.0)
  - Lower duty cycle
  - Faster connection
  - Lower peak power (relaxed RF parameters)

- Compatible with all major platforms (iOS, Android, Windows, Linux)

- Able to discover thousands of devices in local area

- Unlimited number of slaves connected to a master

- Unlimited number of masters

- State of the Art encryption

- Security including privacy/authentication/authorization
Bluetooth Low Energy Branding

2011 Two flavors

- Ultra low power consumption being a pure low energy implementation
- Months to years of lifetime on a standard coin cell battery
- Classic Bluetooth + Bluetooth low energy on a single chip
- These are the hub devices of the Bluetooth ecosystem

2017 Back to one flavor

- An implementation of the Bluetooth core system has only one Primary Controller which may be one of the following configurations:
  - BR/EDR Controller (3.0 and earlier)
  - LE (low energy) Controller (4.0 and newer)
  - Combined BR/EDR Controller portion and LE controller portion into a single Controller (4.0 and newer)

Source: Bluetooth SIG
Bluetooth Low Energy Main Applications

- **FITNESS**
- **SECURITY & PROXIMITY**
- **HEALTHCARE**
- **HOME AUTOMATION**

Technology Tour 2017
Bluetooth Certification

• Is a requirement

• Logo or not, technology is IP of the Bluetooth SIG

• Must be a member of the Bluetooth SIG
  • Adopter: FREE
  • Associate: $7,500 - $35,000 Annual FEE

• Use of pre-certified Components are beneficial
  • Bluetooth Stack
  • Profiles/Services
  • RF Modules

• Listing FEE minimum required
Bluetooth Low Energy Protocol Stack: PHY

- 2.4 GHz ISM Band Transceiver
  - Divided into 40 RF Channels
  - 2 MHz channel Spacing.
  - Two types of channels (3 Advertising and 37 Data)

Source: Bluetooth® SIG
• Bluetooth Low Energy Power Consumption
• Maximize Bluetooth Low Energy Application Throughput
• Voice over Bluetooth Low Energy
• Over The Air (OTA) FW Upgrade over Bluetooth Low Energy
Bluetooth Low Energy Power Consumption
Bluetooth Low Energy Technology

• From bluetooth.org:
  • Bluetooth low energy is the power-version of Bluetooth that was built for the Internet of Things
  • The power-efficiency of Bluetooth with low energy functionality makes it perfect for devices that run for long periods on power sources, such as coin cell batteries or energy-harvesting devices

• Power efficiency is a key aspect, along with:
  • RANGE
  • APPLICATION THROUGHPUT

• Need to look at the whole picture. It is a trade-off balance!
• Many applications require long-range connectivity: asset tracking, home/electronic appliances control, etc…

• Environmental obstacles, causing multipath fading, reduce range from ~30 mt. (line-of-sight) to ~10 mt.

• BlueNRG product family: +8dBm TX output power (NO external PA) guarantee 80 to 100 meters range.

• Is +8dBm against “low energy”? Not necessarily. Look behind the peak current consumption: the key number is the AVERAGE power consumption.
Typical Scenarios

Low throughput
- Short Connection Intervals (min 7.5msecs)
- Active mode Tx/Rx
- Deep sleep mode

High throughput
- Long Connection Intervals (in the orders of secs)
- Active phase
- Peak TX output power counts only here

Active phase typical plot
- mA: Start-up, CPU Active, TX, RX, RX-2-TX, CPU Active, CPU OFF

Technology Tour 2017
Evaluating Average Power Consumption

• Key factors affecting average power numbers:
  • HW operating conditions:
    • High-speed clock start up time
    • Low-speed clock accuracy
    • RAM retention
    • Supply voltage level
    • Power management architectures
    • Deep sleep modes of the radio core
  • Bluetooth low energy protocol application typical use cases:
    • during Advertising: how frequently, how many payload bytes, how many channels
    • during Connection: connection interval length, how many packets per connection event, how many bytes per packet
    • during Scanning: scanning interval and scanning window lengths
Low Power Modes

• Deep sleep mode can represent most of the application time.

• Efficient management of low power modes lowers significantly avg pwr consumption.

### BlueNRG-1 flexible low power architecture

<table>
<thead>
<tr>
<th>Sleeping Mode</th>
<th>Consumption</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>RUNNING</td>
<td>1.9mA</td>
<td>Core running</td>
</tr>
<tr>
<td>CPU HALT</td>
<td>1.5mA</td>
<td>WFI instruction</td>
</tr>
<tr>
<td>WAKETIMER</td>
<td>900nA</td>
<td>GPIOs and Timer Wakeup</td>
</tr>
<tr>
<td>NOTIMER</td>
<td>500nA</td>
<td>GPIOs only Wakeup</td>
</tr>
</tbody>
</table>

Highest pwr

Lowest pwr

Low Power library combines requests coming from the application with the radio operating mode
Context Save/Restore

- Exiting low power modes should NOT result in losing application context and peripheral configurations, and/or radio stack re-initialization.

- When the BlueNRG-1 exits from SLEEPMODE_WAKETIMER and SLEEPMODE_NOTIMER a reset occurs.

- Low Power Library saves peripherals configuration and application context before deep sleep, restoring it upon exiting from low power state:
  - CPU returns to execute the next instruction after the low power function call
  - No need to re-initialize peripheral and radio stack
  - RAM retention is guaranteed
BlueNRG-1 Power Consumption Tool

- Accurate estimate of average current consumption and battery lifetime

1. Select the device
2. Select the type of event
3. Configure the HW operating conditions
4. Configure the application use case scenario
5. Specify the Battery capacity
6. Read the average current and the estimated battery lifetime

Maximize Bluetooth Low Energy Application Throughput
Application Throughput

• Let’s start with obvious: 1 Mega bits per second is just wrong.
  • 1Mbps is at PHY layer BUT overhead is added by the communication protocol.
  • Elements that limit Bluetooth low energy throughput:
    • Maximum packet length
    • Time delays in between packets
    • Overhead bytes in a packet: packet length, data integrity check, and general packet information.

• Protocol theoretical maximum application throughput is ~270-kbps.

• Few devices reach even half of it, due to un-optimized Stacks and poor radio and/or controller design, OS limitations, etc...
In a Master-Slave Bluetooth low energy connection:

- The theoretical maximum application throughput is:
  \[ \text{Throughput}_{\text{kbps}} = \frac{\# \text{Pkts}_\text{ConnEvt} \times (\# \text{Bytes}_\text{Pkt} - \text{Tot}_\text{Overhead}_\text{Pkt})}{\text{Conn}_\text{Interval}} \]
Bluetooth low eEnergy protocol provides 2 types of packets that do NOT require acknowledgement:

- **NOTIFICATIONS**: messages sent from Slave (Server) to Master (Client)
- **WRITES WITHOUT RESPONSE**: messages sent from Master (Client) to Slave (Server)

No acknowledgements allow multiple packets in a single connection interval: throughput is maximized.
• BlueNRG-1 “Throughput demo” provides 2 different scenarios:
  
  • **Unidirectional**: *Notifications* from Server to Client only.
  
  • **Bidirectional**: Simultaneously *Notifications* from Server-to-Client and *Write without response* from Client-to-Server
BlueNRG-1 Navigator Tool

- BlueNRG-1 Navigator: user friendly PC GUI

- Select and run demonstration applications easily, without requiring any extra hardware:
  - BlueNRG-1 Bluetooth low energy demonstration applications
  - BlueNRG-1 peripheral driver examples

- Directly download and run the selected prebuilt application binary images on the BlueNRG-1 platform without a JTAG interface.

- You can select either Bidirectional or Unidirectional prebuilt applications

### Unidirectional scenario

![Message Sequence Chart (MSC) showing ATT_M and ATT_S frames with timestamp and ATT_Handle Value Notification]

<table>
<thead>
<tr>
<th>Frame#</th>
<th>Time</th>
<th>ATT_M</th>
<th>ATT_S</th>
</tr>
</thead>
<tbody>
<tr>
<td>22,730</td>
<td>10:46:57.0902558</td>
<td></td>
<td>ATT_Handle Value Notification [Handle=14]</td>
</tr>
<tr>
<td>22,732</td>
<td>10:46:57.0944811</td>
<td>ATT_Handle Value Notification [Handle=14]</td>
<td></td>
</tr>
<tr>
<td>22,734</td>
<td>10:46:57.094687</td>
<td>ATT_Handle Value Notification [Handle=14]</td>
<td></td>
</tr>
<tr>
<td>22,736</td>
<td>10:46:57.095363</td>
<td>ATT_Handle Value Notification [Handle=14]</td>
<td></td>
</tr>
<tr>
<td>22,738</td>
<td>10:46:57.096831</td>
<td>ATT_Handle Value Notification [Handle=14]</td>
<td></td>
</tr>
<tr>
<td>22,740</td>
<td>10:46:57.096171</td>
<td>ATT_Handle Value Notification [Handle=14]</td>
<td></td>
</tr>
<tr>
<td>22,742</td>
<td>10:46:57.097391</td>
<td>ATT_Handle Value Notification [Handle=14]</td>
<td></td>
</tr>
<tr>
<td>22,744</td>
<td>10:46:57.098367</td>
<td>ATT_Handle Value Notification [Handle=14]</td>
<td></td>
</tr>
</tbody>
</table>

**Average application throughput:** \(~220\text{-}kbps\)
### Bidirectional Scenario

<table>
<thead>
<tr>
<th>Frame#</th>
<th>Time</th>
<th>ATT M</th>
<th>ATT S</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,217</td>
<td>11:17:14.754355</td>
<td>ATT_Write Command</td>
<td>ATT_Handle Value Notification</td>
</tr>
<tr>
<td>1,218</td>
<td>11:17:14.754801</td>
<td>ATT_Write Command</td>
<td>ATT_Handle Value Notification</td>
</tr>
<tr>
<td>1,219</td>
<td>11:17:14.755247</td>
<td>ATT_Write Command</td>
<td>ATT_Handle Value Notification</td>
</tr>
<tr>
<td>1,220</td>
<td>11:17:14.756693</td>
<td>ATT_Write Command</td>
<td>ATT_Handle Value Notification</td>
</tr>
<tr>
<td>1,221</td>
<td>11:17:14.756139</td>
<td>ATT_Write Command</td>
<td>ATT_Handle Value Notification</td>
</tr>
<tr>
<td>1,222</td>
<td>11:17:14.756585</td>
<td>ATT_Write Command</td>
<td>ATT_Handle Value Notification</td>
</tr>
<tr>
<td>1,223</td>
<td>11:17:14.757031</td>
<td>ATT_Write Command</td>
<td>ATT_Handle Value Notification</td>
</tr>
<tr>
<td>1,224</td>
<td>11:17:14.757477</td>
<td>ATT_Write Command</td>
<td>ATT_Handle Value Notification</td>
</tr>
<tr>
<td>1,225</td>
<td>11:17:14.757923</td>
<td>ATT_Write Command</td>
<td>ATT_Handle Value Notification</td>
</tr>
<tr>
<td>1,226</td>
<td>11:17:14.758368</td>
<td>ATT_Write Command</td>
<td>ATT_Handle Value Notification</td>
</tr>
</tbody>
</table>

The scenario involves a message sequence chart (MSC) showing ATT commands and notifications. An average of approximately 170 kbps is observed for both TX packets and RX packets without response.

---

*Technology Tour 2017*
Voice Over Bluetooth Low Energy
• From `bluetooth.org`:
  • One of the most popular applications for Bluetooth has been wireless audio: ... This uses a version of Bluetooth called BR/EDR (basic rate/enhanced data rate) optimized for sending a steady stream of high quality data in a power efficient way.

• Is voice and audio possible using Bluetooth low energy instead?

• BlueVoice library is real-world example of voice and music streaming over a Bluetooth low energy link

  Voice controlled TV remote  Wearables  Smart Home
BlueVoice Architecture: Audio Processing and Transmission

Audio Acquisition → PDM to PCM conversion → Audio Compression → Raw Data

Server - TX

Audio Decompression → Serial Audio Out

USB, I2S, …

Client - RX

symmetrical architecture for bi-directional communication
Transmitter

Microphone acquisition

HP filter + gain correction

ADPCM audio compression

BlueNRG-1 Driver

MEMS Microphones

PDM

8/16 kHz

Motion MEMS

BlueNRG-1

Microphones acquired and sent via Bluetooth Low Energy

Accelerometer and gyroscope data acquired and sent via Bluetooth Low Energy

Included in the BlueNRG-1 DK

Library distributed as middleware

source code example + BlueVoice Library
Architecture 1: BlueMS app

- BlueMS app
- ADPCM audio decompression
- Raw data
- PCM @8 kHz (*)

(*) 16 KHz version is compatible only with selected Android smartphones

Architecture 2: BLUEVOICELINK1

- open.Audio version based on BlueNRG-MS
- BlueNRG-MS
- Network coprocessor
- SPI
- ADPCM audio decompression
- STM32-L0
- STM32-L4
- STM32-F4
- PCM @8/16kHz
BlueVoice Mapping Over Protocol Stack

BlueVoice Vendor Specific Profile

Audio processing

GAP configuration
- Central Unit (Master)
- Peripheral Unit (Slave)

GATT configuration
- Client
- Server

Service
- Characteristic
- Descriptor

Application: BlueVoice Profile (vendor specific)
- Generic Attribute Profile (GATT)
- Generic Access Profile (GAP)
- Attribute Protocol
- Security Manager
- Logical Link Control and Adaptation Protocol
- Host-Controller Interface
- Link Layer
- Direct Test Mode
- Physical Layer

Audio is exported as a Service

Bluetooth Low Energy Stack
BlueVoice Server Attributes

**BlueVoice Service**

- **Service Declaration**
  - *BVS UUID*
  - Type: primary

- **Audio Characteristic**
  - Characteristic Declaration
    - *AudioData UUID*
    - Char properties:
      - Notification
      - Maximum value length: 20 B
  - Characteristic Value
    - Compressed audio data

- **Sync Characteristic**
  - Characteristic Declaration
    - *SyncData UUID*
    - Char properties:
      - Notification
      - Maximum value length: 6 B
  - Characteristic Value
    - Synchronization data
BlueVoice Packet Exchange Sequence

No response is required for Notifications:
- OK for Audio streaming
- Optimal Bandwidth exploitation

Central-to-Peripheral communication

Advertisement Mode

Peripheral-to-Central communication

Central Unit (Master)

Peripheral Unit (Slave)

Discovery Mode

Connection created

Advertisement

Advertise

Connect Request

Notification

Notification

Notification

Write without response

Write without response
Over-The-Air (OTA) FW Upgrade
Over The Air (OTA) FW Upgrade

- What if your device already in the field needs an application FW update?
  - **Over the air (OTA) firmware upgrade** is a protocol that allows a Slave to receive a FW image from a Master over the Bluetooth low energy link (and write it in Flash memory).
  - A Vendor Specific **OTA service** is defined. It coexists with other services used by any application running on the radio stack.

![Diagram of OTA Service]

- **Service Declaration**
  - **OTA UUID**
  - **Type:** primary

- **OTA Service**
  - **OTA bootloader image**
  - **OTA bootloader new image**
  - **OTA bootloader expected image content**
  - **OTA bootloader expected image sequence number**
  - **Lower & Higher bounds of free memory**
  - **Base address and the size of the new FW image**
  - **16-byte data + ctrl info (2-byte block seq nb, 1-byte checksum)**
  - **next expected data block**
Flash Partition Options

- **Reset manager:**
  - jump to valid application

- **OTA service manager:**
  - download new flash image

OTA Reset Manager: **two applications** with OTA Service & Characteristics

OTA Service Manager: **one application** with no OTA Service & Characteristics
## OTA FW Upgrade Properties

<table>
<thead>
<tr>
<th></th>
<th>Lower &amp; higher applications with OTA Reset manager</th>
<th>One a application with OTA Service Manager</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fail safe</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Device always working</td>
<td>Yes (*)</td>
<td>No</td>
<td>(*) In the event of unsuccessful upgrade, the old version of the application will be still operational</td>
</tr>
<tr>
<td>OTA Service &amp; Characteristics</td>
<td>Yes</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>Upgrade policy</td>
<td>Upgradable</td>
<td>Fixed</td>
<td></td>
</tr>
<tr>
<td>Bluetooth Low Energy stack</td>
<td>Upgradable</td>
<td>Upgradable</td>
<td></td>
</tr>
<tr>
<td>Application</td>
<td>Upgradable</td>
<td>Upgradable</td>
<td></td>
</tr>
<tr>
<td>Max Application size</td>
<td>76 kB</td>
<td>98 kB</td>
<td></td>
</tr>
</tbody>
</table>
Flash Layout and Architecture: opt #1

**Higher Application** (with Bluetooth Low Energy OTA service)

- Defines the OTA service in order to provide itself the OTA FW upgrade capability.
- Accepts new application image packets over the air and store them in a specific Flash section (Higher Application storage area).

**Lower Application** (with Bluetooth Low Energy OTA service)

- Defines the OTA service in order to provide itself the OTA FW upgrade capability.
- Accepts new application image packets over the air and store them in a specific Flash section (Lower Application storage area).

Starts after reset and give control to the latest updated Bluetooth Low Energy application.
Flash Layout and Architecture: opt #2

- Application image downloaded through the OTA service manager.
- OTA FW upgrade service is NOT required: needs to simply jump to the OTA service manager (a specific API is provided).
- Always placed at fixed flash address.

- Contains only the OTA service & characteristics to provide OTA firmware upgrade capability.
- Includes the OTA reset manager capability to give control to the latest updated & valid application.
OTA Protocol 1/2

Discovery of the device with OTA FW bootloader service

Connect Request

Connection created

Advertising OTA FW bootloader service UUID 128-bits

Read all OTA service and char handles

Read free space on the target slave device's Flash memory

Write the image base address and size into the Btl new image characteristic

GATT DISC CHAR BY UUID

GATT READ CHAR VALUE

GATT WRITE CHAR VALUE
OTA protocol 2/2

Data transfer: 16 bytes block on the Bootloader new image content characteristic

As slave's internal buffer filled up by 16 byte blocks, it writes into Flash. Notification sent back to Master providing with the block number of the next expected block.

Nb of bytes downloaded on the Slave Flash equal to the image size. OTA protocol writes a validity tag on Slave Flash. OTA Reset/Service Manager jumps to the new application.
Bluetooth Low Energy Roadmap
Flexible Bluetooth Low Energy connectivity solutions: Plug-in for MCU or Programmable Bluetooth Low Energy

<table>
<thead>
<tr>
<th>Scalable MCU + Bluetooth Low Energy bundle</th>
<th>Programmable Bluetooth Low Energy processor</th>
</tr>
</thead>
<tbody>
<tr>
<td>STM32 + BlueNRG-MS</td>
<td>BlueNRG-1</td>
</tr>
<tr>
<td></td>
<td>BlueNRG-2</td>
</tr>
<tr>
<td></td>
<td>BlueNRG-1 / -2</td>
</tr>
</tbody>
</table>

**Scalable Performances with Bluetooth Low Energy Connectivity**

- Pluggable Bluetooth Low Energy architecture
- One Bluetooth Low Energy, Multiple MCU choices
- Full-featured MCU peripherals
- Memory and MIPS scalability
- Package / GPIO flexibility

**BlueNRG-1**
- ARM Cortex-M0 core (32MHz)
- 160kB Flash, 24kB RAM
- Up to 15 GPIO

**BlueNRG-2**
- ARM Cortex-M0 core (32MHz)
- 256kB Flash, 24kB RAM
- Extended Data Packet
- Up to 15 GPIO

**BlueNRG-1 / -2**
- The single-chip solution

- Wireless Sensor Networks (WSN)
- Beacon, Tags and Finders
- Automotive applications
- Smart remote controllers
- Healthcare monitoring
BlueNRG-1 / -2 Bluetooth Low Energy Highlights

Core & Memories
- Cortex-M0 @ 32MHz
- RAM: 2x12KB Ultra Low Leakage
- Flash: 160KB or 256KB (BlueNRG-2)

Peripherals
- Up to 15xGPIO
- 1xSPI, 3xI²C, 1xUART, 1x10-bit ADC
- 16 or 32MHz Xtal and 32KHz for RTC

Extended Operating range
- 1.7 up to 3.6V
- -40 up to +105°C
- AEC-Q100 Automotive Grade

BlueNRG-MS radio
- Up to +8dBm and down to -88dBm
- Rx 7.3 mA and Tx 8.2 mA @ 0dBm
- Sleep current < 1µA

Packages
- BLUENRG-134/-234: WCSP34 2.65x2.65
- BLUENRG-132/-232: QFN32 5.0x5.0
- BLUENRG-132Y: QFN32 with AEC-Q100
- SPBTLE-1S (module): 13 x 11
“STM32 Bluetooth Low Energy Profiles” is a companion tool to show all notifications coming from Bluetooth Low Energy (Bluetooth Low Energy) devices implementing peripheral profiles (e.g. Heart rate, Blood Pressure, Glucose, Health Thermometer, Alert, Proximity, etc.) It supports primarily STM32 Nucleo boards + X-NUCLEO-IDB04A1/X-NUCLEO-IDB05A1 BlueNRG expansion boards running OSXSmartConnPS middleware libraries and examples.
BlueNRG Bluetooth Low Energy
Available Profiles

A growing portfolio of pre-certified profiles

- Alert Notification Client Profile
- Alert Notification Server Profile
- Blood Pressure Profile
- Find Me Locator Profile
- Find Me Target Profile
- Glucose Sensor Profile
- Health Thermometer Profile
- Heart Rate Profile

- HID device
- Phone Alert Profile
- Proximity Monitor Profile
- Proximity Reporter Profile
- Time Server Profile
- Time Client Profile
- A4WP (Alliance 4 Wireless Power)
Bluetooth Low Energy: Conclusion

- Ultra Low power
- Compatible with all major platforms (iOS, Android, Windows, Linux)
- Secure
- Easy to use (already part of most phone and tablets)
- Fast evolving standard: more range, higher data rate, upcoming Mesh and Voice profiles

Bluetooth Low Energy: One Wireless Standard to rule the IoT!
Thank you