| *BLUETOOTH*® DOC | Date / Year-Month-Day<br>2011-09-15 | Approved<br>Adopted | Revision<br>V10r00 | Document No<br>PASP_SPEC |
|---|---|---|---|---|
| Prepared By<br>PUID WG | E-mail Address<br>rd-feedback@bluetooth.org | | | N.B. |

# PHONE ALERT STATUS PROFILE


## Abstract

This profile enables a PUID device to alert its user about the alert status of a phone connected to the PUID device.

# Revision History

| Revision | Date | Comments |
|---|---|---|
| D09r01 | 2011-01-27 | First Draft |
| D09r02 | 2011-03-02 | Updated role name, added BR/EDR |
| D09r03 | 2011-04-26 | Comment resolution in WG |
| D09r04 | 2011-05-13 | Following the latest UCRDD(D05r12) |
| D09r05 | 2011-05-20 | Added connection establishment |
| D09r06 | 2011-05-31 | Added description with HFP to concurrency section |
| D09r07 | 2011-06-11 | Removed BR/EDR and aligned sections 2.4, 5 and 6 with FMP |
| D09r08 | 2011-06-22 | Updated section 5 to latest Proximity. Used section 6 from Koyama's 07_SK, and cleaned up section 7. |
| D09r09 | 2011-06-28 | Comment resolution of comments from Kanji, Terry and Morgan |
| D09r10 | 2011-07-04 | Added Read Characteristic |
| D09r11 | 2011-07-06 | Mandated GAP roles after BARB call. Incorporated comments from Jason (GPA) and Huanchun (GPA/BARB) |
| D09r12 | 2011-07-08 | Editorial in 2.4. Changed to correct Disclaimer |
| V09r00 | 2011-07-26 | Adopted as PS by the Bluetooth SIG Board of Directors |
| D10r01 | 2011-08-09 | First draft D10 |
| D10r02 | 2011-08-10 | Added Service to the service UUID name |
| D10r03 | 2011-08-17 | Added section 2.5, transport dependencies, removed PS disclaimer, fixed service reference to v1.0, Resolved comments from Anindya. Changed some template explanation of GAP modes. |
| D10r04 | 2011-08-26 | Responded to Comments from Joakim. |
| V10r00 | 2011-09-15 | Adopted by Bluetooth SIG Board of Directors |

# Contributors

| Name | Company |
|---|---|
| Koyama Shunsuke | Seiko Epson |
| Satoshi Oshiyama | Seiko Epson |
| Sadao Nagashima | Casio |
| Daisuke Matsuoh | Citizen |
| Frank Berntsen | Nordic Semiconductor |

# Disclaimer and Copyright Notice

The copyright in this specification is owned by the Promoter Members of Bluetooth® Special Interest Group (SIG), Inc. ("Bluetooth SIG"). Use of these specifications and any related intellectual property (collectively, the "Specification"), is governed by the Promoters Membership Agreement among the Promoter Members and Bluetooth SIG (the "Promoters Agreement"), certain membership agreements between Bluetooth SIG and its Adopter and Associate Members (the "Membership Agreements") and the Bluetooth Specification Early Adopters Agreements (1.2 Early Adopters Agreements) among Early Adopter members of the unincorporated Bluetooth SIG and the Promoter Members (the "Early Adopters Agreement"). Certain rights and obligations of the Promoter Members under the Early Adopters Agreements have been assigned to Bluetooth SIG by the Promoter Members.

Use of the Specification by anyone who is not a member of Bluetooth SIG or a party to an Early Adopters Agreement (each such person or party, a "Member") is prohibited. The legal rights and obligations of each Member are governed by their applicable Membership Agreement, Early Adopters Agreement or Promoters Agreement. No license, express or implied, by estoppel or otherwise, to any intellectual property rights are granted herein.

Any use of the Specification not in compliance with the terms of the applicable Membership Agreement, Early Adopters Agreement or Promoters Agreement is prohibited and any such prohibited use may result in termination of the applicable Membership Agreement or Early Adopters Agreement and other liability permitted by the applicable agreement or by applicable law to Bluetooth SIG or any of its members for patent, copyright and/or trademark infringement.

**THE SPECIFICATION IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, SATISFACTORY QUALITY, OR REASONABLE SKILL OR CARE, OR ANY WARRANTY ARISING OUT OF ANY COURSE OF DEALING, USAGE, TRADE PRACTICE, PROPOSAL, SPECIFICATION OR SAMPLE.**

Each Member hereby acknowledges that products equipped with the *Bluetooth* technology ("*Bluetooth* products") may be subject to various regulatory controls under the laws and regulations of various governments worldwide. Such laws and regulatory controls may govern, among other things, the combination, operation, use, implementation and distribution of *Bluetooth* products. Examples of such laws and regulatory controls include, but are not limited to, airline regulatory controls, telecommunications regulations, technology transfer controls and health and safety regulations. Each Member is solely responsible for the compliance by their *Bluetooth* Products with any such laws and regulations and for obtaining any and all required authorizations, permits, or licenses for their *Bluetooth* products related to such regulations within the applicable jurisdictions. Each Member acknowledges that nothing in the Specification provides any information or assistance in connection with securing such compliance, authorizations or licenses. **NOTHING IN THE SPECIFICATION CREATES ANY WARRANTIES, EITHER EXPRESS OR IMPLIED, REGARDING SUCH LAWS OR REGULATIONS.**

**ALL LIABILITY, INCLUDING LIABILITY FOR INFRINGEMENT OF ANY INTELLECTUAL PROPERTY RIGHTS OR FOR NONCOMPLIANCE WITH LAWS, RELATING TO USE OF THE SPECIFICATION IS EXPRESSLY DISCLAIMED. BY USE OF THE SPECIFICATION, EACH MEMBER EXPRESSLY WAIVES ANY CLAIM AGAINST BLUETOOTH SIG AND ITS PROMOTER MEMBERS RELATED TO USE OF THE SPECIFICATION.**

Bluetooth SIG reserve the right to adopt any changes or alterations to the Specification as it deems necessary or appropriate.

## Document Terminology

The Bluetooth SIG has adopted Section 13.1 of the IEEE Standards Style Manual, which dictates use of the words ``shall'', ``should'', ``may'', and ``can'' in the development of documentation, as follows:

The word *shall* is used to indicate mandatory requirements strictly to be followed in order to conform to the standard and from which no deviation is permitted (*shall* equals *is required to*).

The use of the word *must* is deprecated and shall not be used when stating mandatory requirements; *must* is used only to describe unavoidable situations.

The use of the word *will* is deprecated and shall not be used when stating mandatory requirements; *will* is only used in statements of fact.

The word *should* is used to indicate that among several possibilities one is recommended as particularly suitable, without mentioning or excluding others; or that a certain course of action is preferred but not necessarily required; or that (in the negative form) a certain course of action is deprecated but not prohibited (*should* equals *is recommended that*).

The word *may* is used to indicate a course of action permissible within the limits of the standard (*may* equals *is permitted*).

The word *can* is used for statements of possibility and capability, whether material, physical, or causal (*can* equals *is able to*).

# Table of Contents

# 1 Introduction

The Phone Alert Status profile is used to obtain the Phone Alert Status exposed by the Phone Alert Status service in the peer device. The information of Alert Status and Ringer Setting of a phone can be received and changed by the Phone Alert Status service. This profile also enables the device to configure ringer status on the peer device.

## 1.1 Profile Dependencies

This profile is compatible with any *Bluetooth* core specification Host that includes the Generic Attribute Profile (GATT).

## 1.2 Conformance

If conformance to this profile is claimed, all capabilities indicated as mandatory for this profile shall be supported in the specified manner (process mandatory). This also applies for all optional and conditional capabilities for which support is indicated. All mandatory capabilities, and optional and conditional capabilities for which support is indicated, are subject to verification as part of the Bluetooth qualification program.

# 2 Configuration

## 2.1 Roles

The Profile defines two roles Phone Alert Server and Phone Alert Client. The Phone Alert Server is the device that originates the alerts and the Phone Alert Client is the device that receives the alerts and alerts the user.

- The Phone Alert Server shall be a GATT server.
- The Phone Alert Client shall be a GATT client.

## 2.2 Roles/Service Relationship

Figure 2.1 shows the relationships between service and the two profile roles.

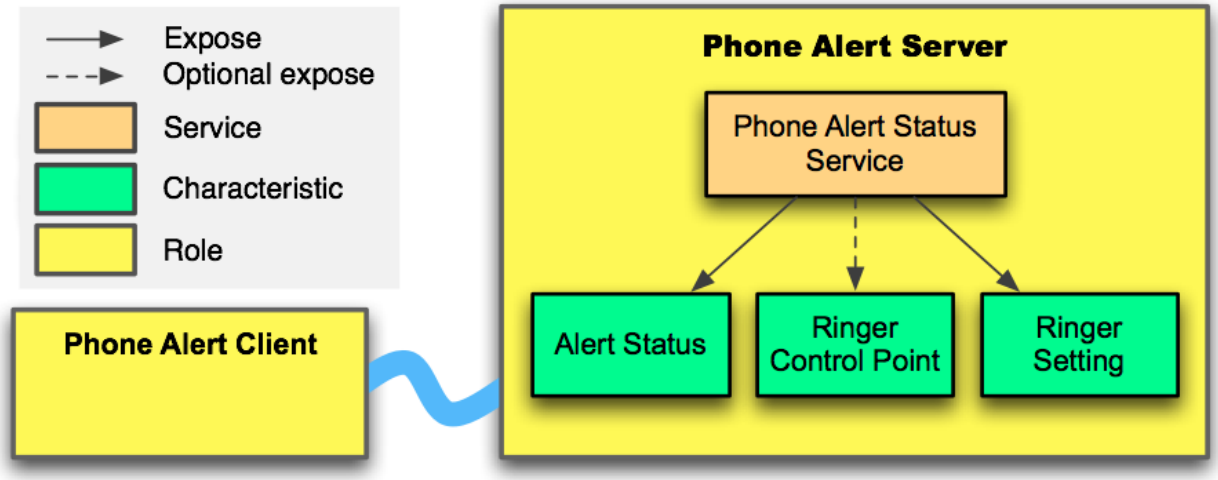


*Figure 2.1: Role and service relationships*

## 2.3 Concurrency

The Phone Alert Status profile may run concurrently with other profiles.

In multi-profile scenarios (for example when the server device also runs HFP with another device), the Phone Alert Client device should act as an extension of the phone's UI and thus the interaction of the Phone Alert Status device with the phone should be the same as what happens on the phone in any given situation. This way whatever can or cannot be done on the phone's UI, when in a multi-profile use case, also applies to the Phone Alert Status device.

## 2.4 Topology

The Phone Alert Server shall implement the GAP Central role and may implement the GAP Peripheral role. The Phone Alert Client shall implement the GAP Peripheral role and may implement the GAP Central role.

## 2.5 Transport Dependencies

This profile shall operate over an LE transport.

# 3 Phone Alert Server Requirements

This profile does not impose any additional requirements on any of the instances of the Phone Alert Status Service beyond those defined by the specifications.

## 3.1 Phone Alert Status Service

A device implementing the Phone Alert Status profile in the Server role shall have one instance of Phone Alert Status service.

# 4 Phone Alert Client Requirements

This section describes the procedure requirements for a Phone Alert Client.

| | Procedure | Ref. | Support in Phone Alert Client |
|---|---|---|---|
| 1. | Service Discovery | 4.1 | M |
| 2. | Characteristic Discovery | 4.2 | M |
| 3. | Read Phone Alert Status | 4.3 | M |
| 4. | Receive notification on Phone Alert Status | 4.4 | M |
| 5. | Request to notify when Alert Status changes | 4.5 | M |
| 6. | Read Ringer Setting | 4.6 | M |
| 7. | Receive notification on Ringer Setting | 4.7 | O |
| 8. | Request to notify when Ringer Setting changes | 4.8 | C.1 |
| 9. | Set the peer to “Ringer Silent” or “Ringer Normal” | 4.9 | O |
| 10. | Mute the Ringer once | 4.10 | O |
| 11. | Check the Alert Status after Connection Establishment | 4.11 | M |
| C.1: Mandatory if notifications of Ringer Setting is supported, otherwise optional | | | |

*Table 4.1: Procedure requirements for Phone Alert client*

## 4.1 Service Discovery

The Phone Alert Client shall perform service discovery using the GATT *Discover All Primary Services* sub-procedure or the GATT *Discover Primary Services by Service UUID* sub-procedure using «Phone Alert Status Service» for the service UUID.

## 4.2 Characteristic Discovery

The GATT sub-procedure *Discover All Characteristic of a Service* or the GATT sub-procedure *Discover Characteristics by UUID* shall be used to discover the characteristics of the Phone Alert Status service.

The GATT *Discover All Characteristic Descriptors* sub-procedure shall be used to discover the *Client Characteristic Configuration* descriptor.

## 4.3 Read Phone Alert Status

Phone Alert Client shall read the Phone Alert Status on the Phone Alert Server.

## 4.4 Receive notification on Phone Alert Status

Phone Alert Client shall receive Phone Alert Status notification from Phone Alert Server.

## 4.5 Request to notify when the Alert Status changes

The Phone Alert Client shall control the configuration of notifications of the Alert Status characteristic by using the GATT *Write Characteristic Descriptors* sub-procedure to write its Client Characteristic Configuration.

## 4.6 Read Ringer Setting

Phone Alert Client shall read the Ringer Setting Characteristic on the Phone Alert Server.

## 4.7 Receive notification on Ringer Setting

Phone Alert Client shall receive Ringer Setting notifications from Phone Alert Server.

## 4.8 Request to Notify when the Ringer Setting Changes

The Phone Alert Client may control the configuration of notifications of the Ringer Setting characteristic by using the GATT *Write Characteristic Descriptors* sub-procedure to write its Client Characteristic Configuration descriptor.

## 4.9 Set the Peer to “Ringer Silent” or “Ringer Normal” State

To request the Phone Alert Server to change its state to “Ringer Silent”, the Phone Alert Client shall write one of the two following commands into the Ringer Control Point.

- To request the peer to go to the “Ringer Silent” state, the command “Set Silent Mode” shall be written.
- To request the peer to go to the “Ringer Normal” state, the command “Cancel Silent Mode” shall be written.

## 4.10 Mute the Ringer Once

To request the Phone Alert Server to mute the Ringer once, the Phone Alert Client shall write the command “Mute Once” into the Ringer Control Point.

## 4.11 Check the Alert Status after Connection Establishment

The Phone Alert Client shall read the value of the Alert Status characteristic after connection setup. The Phone Alert Client should then alert the user based on the value of the Alert Status characteristic.

# 5 Connection Establishment

This section describes the connection establishment procedures used by a Phone Alert Client and Phone Alert Server. Since there are no topology restrictions imposed by this profile, the procedures are described in terms of GAP Peripheral role (referred to as the Peripheral) and GAP Central role (referred to as the Central).

## 5.1 GAP Peripheral Role Connection Establishment

### 5.1.1 Device Discovery

The Peripheral shall enter a GAP Limited Discoverable Mode when establishing an initial connection. The $T_{GAP}$ (lim_adv_timeout) used during GAP Limited Discoverable Mode may be larger than the value specified in the Section 16, Appendix A in the GAP specification [1], but the value shall be less than or equal to 180 seconds.

### 5.1.2 Connection Procedure for Unbonded Devices

This procedure is used for device discovery and connection establishment when the Peripheral connects to a Central to which it is not bonded. This procedure is initiated by user interaction (like activating the device by battery insertion).

It is recommended that the Peripheral advertise using the parameters in Table 5.1. The interval values in the first row are designed to attempt fast connection during the first 30 seconds; however, if a connection is not established within that time, the interval values in the second row are designed to reduce power consumption for devices that continue to advertise.

| Advertising Duration | Parameter | Value |
|---|---|---|
| First 30 seconds (fast connection) | Advertising Interval | 20 ms to 30 ms |
| After 30 seconds (reduced power) | Advertising Interval | 1 s to 2.5 s |

*Table 5.1: Recommended advertising interval values*

The advertising interval and time to perform advertising should be configured with consideration for user expectations of connection establishment time.

The Peripheral shall accept any valid values for connection interval and connection latency set by the Central until service discovery, bonding, and encryption setup is complete. Only after that should the Peripheral change to the preferred connection parameters that best suits the use case.

If a connection is not established within a time limit defined by the Peripheral, the Peripheral may exit the GAP connectable mode.

After bonding the Peripheral should write the *Bluetooth* address of the Central in the Peripheral controller's white list and set the Peripheral controller's advertising filter policy to 'process scan and connection requests only from devices in the White List'.

### 5.1.3 Connection Procedure for Bonded Devices

This procedure is used after the Peripheral has bonded with the Central device using the connection procedure in Section 5.1.2 when the user initiates a connection.

A Peripheral shall enter the GAP *Undirected Connectable Mode* when commanded by the user to initiate a connection to a Central device.

The Peripheral should use the advertising filter policy configured when bonded using the connection procedure in Section 5.1.2.

The Peripheral should use the recommended advertising interval values shown in Table 5.1.

The advertising interval and time to perform advertising should be configured with consideration for user expectations of connection establishment time.

The Peripheral shall accept any valid values for connection interval and connection latency set by the Central until service discovery and encryption setup is complete. Only after that should the Peripheral change to the preferred connection parameters that best suits its use case.

If a connection is not established within a time limit defined by the Peripheral, the Peripheral may exit the GAP connectable mode.

### 5.1.4 Link Loss Reconnection

When a connection is terminated due to link loss a Peripheral should attempt to reconnect to the Central by using the procedures described in 5.1.2 or 5.1.3.

## 5.2 GAP Central Role Connection Establishment

### 5.2.1 Device Discovery

The Central should use the GAP Limited Discovery Procedure to discover a Peripheral.

### 5.2.2 Connection Procedure for Unbonded Devices

This procedure is used for connection establishment when the Central connects to a Peripheral to which it is not bonded. This procedure is normally initiated by user interaction.

A Central may use one of the following GAP connection establishment procedures based on its connectivity requirements:

- *General Connection Establishment Procedure*. The Central may use this procedure when it requires connecting to one or more Peripheral devices. This procedure allows a Central to connect to a Peripheral discovered during a scan without using the white list.
- *Direct Connection Establishment Procedure*. The Central may use this procedure when it requires connecting to a single Peripheral.

- *Auto Connection Establishment Procedure*. The Central may use this procedure when it requires connecting to one or more Peripheral devices. This procedure will automatically connect to a Peripheral in the white list.
- *Selective Connection Establishment Procedure*. The Central may use this procedure when it requires connecting to one or more Peripheral devices. This procedure allows a Central to connect to a Peripheral discovered during a scan while using the white list.

A Central should use the recommended scan interval and scan window values shown in Table 5.2. For the first 30 seconds (or optionally continuously for mains powered devices), the Central should use the first scan window / scan interval pair to attempt fast connection. However, if a connection is not established within that time, the Central should switch to one of the other scan window / scan interval options as defined below to reduce power consumption.

| Scan Duration | Parameter | Value |
|---|---|---|
| First 30 seconds (fast connection) | Scan Interval | 30 ms to 60 ms* |
| | Scan Window | 30 ms |
| After 30 seconds (reduced power) - Option 1 | Scan Interval | 1.28 s |
| | Scan Window | 11.25 ms |
| After 30 seconds (reduced power) - Option 2 | Scan Interval | 2.56 s |
| | Scan Window | 11.25 ms |

*Table 5.2: Recommended scan interval and scan window values*

* A scan interval of 60ms is recommended when the Central is supporting other operations to provide a 50% scan duty cycle versus 100% scan duty cycle.

Option 1 in the table above uses the same background scanning interval used in BR/EDR so the power consumption for LE will be similar to the power consumption used for background scanning on BR/EDR. Option 2 uses a larger background scanning interval (e.g. twice as long) than used in BR/EDR so the power consumption for LE will be less than the power consumption used for background scanning on BR/EDR. Connection times during background scanning will be longer with Option 2.

After bonding, the Central should write the *Bluetooth* address of the Peripheral in the Central controller's white list and set the Central controller's initiator filter policy to 'process connectable advertisement packets'.

### 5.2.3 Connection Procedure for Bonded Devices

This procedure is used after the Central has bonded with the Peripheral using the connection procedure in Section 5.2.2 and the user initiates a connection.

A Central may use one of the following GAP connection establishment procedures based on its connectivity requirements:

- *General Connection Establishment Procedure*. The Central may use this procedure when it requires connecting to one or more peripheral devices. This procedure allows a Central to connect to a Peripheral discovered during a scan without using the white list.

- *Direct Connection Establishment Procedure*. The Central may use this procedure when it requires connecting to a single peripheral.
- *Auto Connection Establishment Procedure*. The Central may use this procedure when it requires connecting to one or more peripheral devices. This procedure will automatically connect to a Peripheral in the white list.
- *Selective Connection Establishment Procedure*. The Central may use this procedure when it requires connecting to one or more peripheral devices. This procedure allows a Central to connect to a peripheral discovered during a scan while using the white list.

The Central should use the recommended scan interval and scan window values shown in Table 5.2. For the first 30 seconds (or optionally continuously for mains powered devices), the Central should use the first scan window / scan interval pair to attempt fast connection. However, if a connection is not established within that time, the Central should switch to one of the other scan window / scan interval options as defined below to reduce power consumption.

The Central should use a scan window and scan interval suitable to its power and connection time requirements. Increasing the scan window increases the power consumption, but decreases the connection time.

The scan interval and scan window should be configured with consideration for user expectations of connection establishment time.

The Central shall start encryption after each connection creation to verify the status of the bond. If encryption fails upon connection establishment (i.e., The bond no longer exists), the Central must, after user interaction, re-bond, perform service discovery (unless the Central had previously determined that the Peripheral did not have the «Service Changed» characteristic) and reconfigure the Peripheral before using any of the services referenced by this profile in case the configuration was altered or lost.

### 5.2.4 Link Loss Reconnection

When a connection is terminated due to link loss a Central should attempt to reconnect to the Peripheral using any of the GAP connection procedures and using procedures described in sections 5.2.2 or 5.2.3 .

### 5.2.5 Fast Connection Interval

To avoid very long service discovery and encryption setup times, the Central should use the connection intervals defined in Table 5.3 in the connection request.

| Parameter | Value |
|---|---|
| Minimum Connection Interval | 50 ms |
| Maximum Connection Interval | 70 ms |

*Table 5.3: Recommended connection interval values*

At any time a key refresh or encryption setup is required, for example to perform key refresh, this should be preceded with a connection parameter update to the minimum and maximum connection interval values in Table 5.3 and a latency of zero. This fast

connection interval should be maintained as long as low latency is required. After that, it should switch to the preferred connection parameters as decided by the Peripheral using the GAP *Connection Parameter Update* procedure.

# 6 Security Considerations

This section describes the security requirements for a Phone Alert Client and Phone Alert Server. Since there are no topology restrictions imposed by this profile, the requirements are described in terms of GAP Peripheral Role (referred to as the Peripheral) and GAP Central Role (referred to as the Central).

The Peripheral shall support LE Security Mode 1 and Security Level 2 or 3. The Peripheral should use the SM Slave Security Request procedure only when bonded with the Central to inform the Central of its security requirements.

The Central shall support LE Security Mode 1 and Security Level 2 and 3. The Central should accept the LE Security Mode and Security Level combination requested by the Peripheral.

# 7 GATT Interoperability Requirements

The following GATT sub-procedures are required to be implemented for Phone Alert Client profile role.

| GATT Sub-Procedure | Phone Alert Client |
|---|---|
| Discover All Primary Services | C.1 |
| Discovery Primary Services by Service UUID | C.1 |
| Discover All Characteristic of a Service | C.2 |
| Discover Characteristic by UUID | C.2 |
| Discover All Characteristic Descriptors | M |
| Read Characteristic Value | M |
| Write Characteristic Descriptors | M |
| Notification | M |
| C.1: The Phone Alert Client shall either support *Discover All Primary Services* sub-procedure OR *Discovery Primary Services by Service UUID* sub-procedure. | |
| C.2: The Phone Alert Client shall either support *Discovery All Characteristic of a Service* sub-procedure OR *Read Characteristic Using UUID* sub-procedure. | |

# 8 Acronyms and Abbreviations

| Acronyms and Abbreviations | Meaning |
|---|---|
| ATT | Attribute Protocol |
| BR/EDR | Basic Rate / Enhanced Data Rate |
| GAP | Generic Access Profile |
| GATT | Generic Attribute Profile |
| LE | Low Energy |
| UUID | Universally Unique Identifier |

# 9 References

[1] *Bluetooth* Core Specification v4.0
[2] Phone Alert Status Service v1.0