

Bluetooth[®] Low Energy Protocol Stack

API Reference Manual: PXP

Renesas MCU

Target Device

RL78/G1D

All information contained in these materials, including products and product specifications, represents information on the product at the time of publication and is subject to change by Renesas Electronics Corp. without notice. Please review the latest information published by Renesas Electronics Corp. through various means, including the Renesas Electronics Corp. website (<http://www.renesas.com>).

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
3. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.
"Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots etc.
"High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; and safety equipment etc.

Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantations etc.), or may cause serious property damages (nuclear reactor control systems, military equipment etc.). You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application for which it is not intended. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for which the product is not intended by Renesas Electronics.
6. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You should not use Renesas Electronics products or technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. When exporting the Renesas Electronics products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document, Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
11. This document may not be reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Handling of Unused Pins

Handle unused pins in accordance with the directions given under Handling of Unused Pins in the manual.

- The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible. Unused pins should be handled as described under Handling of Unused Pins in the manual.

2. Processing at Power-on

The state of the product is undefined at the moment when power is supplied.

- The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the moment when power is supplied.
In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the moment when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the moment when power is supplied until the power reaches the level at which resetting has been specified.

3. Prohibition of Access to Reserved Addresses

Access to reserved addresses is prohibited.

- The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.

4. Clock Signals

After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has stabilized.

- When the clock signal is generated with an external resonator (or from an external oscillator) during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Moreover, when switching to a clock signal produced with an external resonator (or by an external oscillator) while program execution is in progress, wait until the target clock signal is stable.

5. Differences between Products

Before changing from one product to another, i.e. to a product with a different part number, confirm that the change will not lead to problems.

- The characteristics of an MPU or MCU in the same group but having a different part number may differ in terms of the internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

How to Use This Manual

1. Purpose and Target Readers

This manual describes the API (Application Program Interface) of the Proximity profile (PXP) of the Bluetooth Low Energy protocol stack (BLE software), which is used to develop Bluetooth applications that incorporate the Renesas Bluetooth low energy microcontroller RL78/G1D. It is intended for users designing application systems incorporating this software. A basic knowledge of microcontrollers and Bluetooth low energy is necessary in order to use this manual.

Related documents

The related documents indicated in this publication may include preliminary versions. However, preliminary versions are not marked as such.

Document Name	Document No.
Bluetooth Low Energy Protocol Stack	
User's Manual	R01UW0095E
API Reference Manual: Basics	R01UW0088E
API Reference Manual: FMP	R01UW0089E
API Reference Manual: PXP	This manual
API Reference Manual: HTP	R01UW0091E
API Reference Manual: BLP	R01UW0092E
API Reference Manual: HOGP	R01UW0093E
API Reference Manual: ScPP	R01UW0094E
API Reference Manual: HRP	R01UW0097E
API Reference Manual: CSCP	R01UW0098E
API Reference Manual: CPP	R01UW0099E
API Reference Manual: GLP	R01UW0103E
API Reference Manual: TIP	R01UW0106E
API Reference Manual: RSCP	R01UW0107E
API Reference Manual: ANP	R01UW0108E
API Reference Manual: PASP	R01UW0109E
API Reference Manual: LNP	R01UW0113E
Application Note: Sample Program	R01AN1375E
Application Note: rBLE Command Specification	R01AN1376E

List of Abbreviations and Acronyms

Abbreviation	Full Form	Remark
ANP	Alert Notification Profile	
ANS	Alert Notification Service	
API	Application Programming Interface	
ATT	Attribute Protocol	
BAS	Battery Service	
BB	Base Band	
BD_ADDR	Bluetooth Device Address	
BLE	Bluetooth low energy	
BLP	Blood Pressure Profile	
BLS	Blood Pressure Service	
CPP	Cycling Power Profile	
CPS	Cycling Power Service	
CSCP	Cycling Speed and Cadence Profile	
CSCS	Cycling Speed and Cadence Service	
CSRK	Connection Signature Resolving Key	
CTS	Current Time Service	
DIS	Device Information Service	
EDIV	Encrypted Diversifier	
FMP	Find Me Profile	
GAP	Generic Access Profile	
GATT	Generic Attribute Profile	
GLP	Glucose Profile	
GLS	Glucose Service	
HCI	Host Controller Interface	
HID	Human Interface Device	
HIDS	HID Service	
HOGP	HID over GATT Profile	
HRP	Heart Rate Profile	
HRS	Heart Rate Service	
HTP	Health Thermometer Profile	
HTS	Health Thermometer Service	
IAS	Immediate Alert Service	
IRK	Identity Resolving Key	
L2CAP	Logical Link Control and Adaptation Protocol	
LE	Low Energy	

Abbreviation	Full Form	Remark
LL	Link Layer	
LLS	Link Loss Service	
LNP	Location and Navigation Profile	
LNS	Location and Navigation Service	
LTK	Long Term Key	
MCU	Micro Controller Unit	
MITM	Man-in-the-middle	
MTU	Maximum Transmission Unit	
NDCS	Next DST Change Service	
OOB	Out of Band	
OS	Operating System	
PASP	Phone Alert Status Profile	
PASS	Phone Alert Status Service	
PXP	Proximity Profile	
RF	Radio Frequency	
RSCP	Running Speed and Cadence Profile	
RSCS	Running Speed and Cadence Service	
RSSI	Received Signal Strength Indication	
RTUS	Reference Time Update Service	
ScPP	Scan Parameters Profile	
ScPS	Scan Parameters Service	
SM	Security Manager	
SMP	Security Manager Protocol	
STK	Short Term Key	
TIP	Time Profile	
TK	Temporary Key	
TPS	Tx Power Service	
UART	Universal Asynchronous Receiver Transmitter	
UUID	Universal Unique Identifier	

Abbreviation	Full Form	Remark
APP	Application	
CSI	Clocked Serial Interface	
IIC	Inter-Integrated Circuit	
RSCIP	Renesas Serial Communication Interface Protocol	
VS	Vendor Specific	

All trademarks and registered trademarks are the property of their respective owners.

Bluetooth is a registered trademark of Bluetooth SIG, Inc. U.S.A.

EEPROM is a trademark of Renesas Electronics Corporation.

Windows, Windows NT and Windows XP are registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

PC/AT is a trademark of International Business Machines Corporation.

Contents

1. Overview.....	1
2. Common Definitions	2
2.1 Service Definitions	2
2.2 Status Definitions.....	4
3. Proximity Profile.....	5
3.1 Definitions	5
3.2 Functions	11
3.2.1 RBLE_PXP_Reporter_Enable.....	12
3.2.2 RBLE_PXP_Reporter_Disable.....	12
3.2.3 RBLE_PXP_Monitor_Enable.....	13
3.2.4 RBLE_PXP_Monitor_Disable.....	14
3.2.5 RBLE_PXP_Monitor_Get_Alert_Level.....	14
3.2.6 RBLE_PXP_Monitor_Set_Alert_Level.....	14
3.2.7 RBLE_PXP_Monitor_Get_Tx_Power	15
3.3 Events	16
3.3.1 RBLE_PXP_EVENT_REPORTER_ENABLE_COMP.....	17
3.3.2 RBLE_PXP_EVENT_REPORTER_DISABLE_COMP.....	17
3.3.3 RBLE_PXP_EVENT_REPORTER_ALERT_IND.....	17
3.3.4 RBLE_PXP_EVENT_REPORTER_COMMAND_DISALLOWED_IND	17
3.3.5 RBLE_PXP_EVENT_MONITOR_ENABLE_COMP	18
3.3.6 RBLE_PXP_EVENT_MONITOR_DISABLE_COMP	18
3.3.7 RBLE_PXP_EVENT_MONITOR_ERROR_IND	19
3.3.8 RBLE_PXP_EVENT_MONITOR_READ_CHAR_RESPONSE.....	19
3.3.9 RBLE_PXP_EVENT_MONITOR_WRITE_CHAR_RESPONSE.....	19
3.3.10 RBLE_PXP_EVENT_MONITOR_COMMAND_DISALLOWED_IND	19
3.4 Message Sequence Chart	20

4. Notes	22
Appendix A How to Read Definition Tables.....	23
Appendix B Referenced Documents	25
Appendix C Terminology	26

1. Overview

This manual describes the API (Application Program Interface) of the Proximity profile (PXP) of the Bluetooth Low Energy protocol stack (BLE software), which is used to develop Bluetooth applications that incorporate Renesas Bluetooth low energy microcontroller RL78/G1D.

For details about the organization and features of BLE software, see the Bluetooth Low Energy Protocol Stack User's Manual.

2. Common Definitions

This section describes the definitions common to the API of each profile.

2.1 Service Definitions

This section describes the common definitions of services used by the API of multiple profiles.

- Declaration of enumerated type for alert level

```
enum RBLE_SVC_ALT_LVL_enum {
    RBLE_SVC_ALERT_NONE = 0x00,          No alert
    RBLE_SVC_ALERT_MILD,                  Mild alert
    RBLE_SVC_ALERT_HIGH                   High alert
};
```

- Declaration of enumerated type for PnP ID characteristic vendor ID field

```
enum RBLE_SVC_PNP_VENDOR_ID_enum {
    RBLE_SVC_SIG_ASSIGNED_ID = 0x01,      Vendor ID assigned by Bluetooth SIG
    RBLE_SVC_USB_ASSIGNED_ID           Vendor ID assigned by USB Implementer's
                                        Forum
};
```

- Declaration of enumerated type for Name Space field of Characteristic Presentation Format descriptor

```
enum RBLE_SVC_PRESEN_NAMESPASE_enum {
    RBLE_SVC_NAMESPACE_SIG = 0x01,        Defined by Bluetooth SIG
};
```

- Declaration of enumerated type for security level of Service

```
enum RBLE_SVC_SEC_LVL_enum {
    RBLE_SVC_SEC_NONE = 0x01,             No security
    RBLE_SVC_SEC_UNAUTH = 0x02,           Require unauthenticated pairing
    RBLE_SVC_SEC_AUTH = 0x04,             Require authenticated pairing
    RBLE_SVC_SEC_AUTZ = 0x08,             Require authorization
    RBLE_SVC_SEC_ENC = 0x10               Require encryption
};
```

- Declaration of enumerated type for connection types

```
enum RBLE_PRF_CON_enum {
    RBLE_PRF_CON_DISCOVERY = 0x00,        Configuration connection performed
                                           when connecting for the first time
    RBLE_PRF_CON_NORMAL           Normal connection performed when
                                           connecting for the second and
                                           subsequent times
};
```

- Declaration of enumerated type for client configuration characteristic value

```
enum RBLE_PRF_CLIENT_CONFIG_enum {  
    RBLE_PRF_STOP_NTFFIND = 0x00,           Stop notification or indication of  
                                              characteristic value.  
    RBLE_PRF_START_NTF,                     Start notification of  
                                              characteristic value.  
    RBLE_PRF_START_IND                       Start indication of  
                                              characteristic value.  
};
```

- Declaration of enumerated type for server configuration characteristic value

```
enum RBLE_PRF_SERVER_CONFIG_enum {  
    RBLE_PRF_STOP_BRD = 0x00,               Stop broadcast of characteristic value.  
    RBLE_PRF_START_BRD                      Start broadcast of characteristic value.  
};
```

2.2 Status Definitions

This section describes the status definitions used by the API of each profile.

- Declaration of enumerated type for rBLE status

```
enum RBLE_STATUS_enum {
    RBLE_OK = 0x00,
    RBLE_PRF_ERR_INVALID_PARAM = 0x90,

    RBLE_PRF_ERR_INEXISTENT_HDL,

    RBLE_PRF_ERR_STOP_DISC_CHAR_MISSING,
    RBLE_PRF_ERR_MULTIPLE_IAS,
    RBLE_PRF_ERR_INCORRECT_PROP,
    RBLE_PRF_ERR_MULTIPLE_CHAR,
    RBLE_PRF_ERR_NOT_WRITABLE,
    RBLE_PRF_ERR_NOT_READABLE,
    RBLE_PRF_ERR_REQ_DISALLOWED,
    RBLE_PRF_ERR_NTF_DISABLED,
    RBLE_PRF_ERR_IND_DISABLED,
    RBLE_PRF_ERR_ATT_NOT_SUPPORTED,
};
```

	Normal operation
	Invalid parameter specified for setting or acquiring a characteristic value
	Invalid handle specified for setting or acquiring a characteristic value
	The characteristic value is missing.
	Multiple IASs exist.
	Incorrect property
	Multiple characteristic values exist.
	Writing is not permitted.
	Reading is not permitted.
	Requesting is not permitted.
	Notification is disabled.
	Indication is disabled.
	The characteristic value is not supported.

Note: Statuses other than the above are described in *API Reference Manual: Basics*.

3. Proximity Profile

This section describes the API of the Proximity profile. The Proximity profile defines the behavior when a device moves away from a peer device so that the connection is dropped or the path loss increases above a preset level, causing an immediate alert to be sent to the user.

3.1 Definitions

This section describes the definitions used by the API of the Proximity profile.

- Declaration of enumerated type for PXP event types

```
enum RBLE_PXPM_EVENT_TYPE_enum {
    RBLE_PXP_EVENT_REPORTER_ENABLE_COMP = 0x01,           Reporter enable completion
                                                           event
                                                           (Parameter: report_enable)
    RBLE_PXP_EVENT_REPORTER_DISABLE_COMP,                 Reporter disable completion
                                                           event
                                                           (Parameter: report_disable)
    RBLE_PXP_EVENT_REPORTER_ALERT_IND,                     Reporter alert level
                                                           indication event
                                                           (Parameter: report_alert_ind)
    RBLE_PXP_EVENT_REPORTER_COMMAND_DISALLOWED_IND,        Reporter command disallowed
                                                           indication event
                                                           (Parameter: cmd_disallowed_ind)
    RBLE_PXP_EVENT_MONITOR_ENABLE_COMP = 0x81,             Monitor enable completion
                                                           event
                                                           (Parameter: monitor_enable)
    RBLE_PXP_EVENT_MONITOR_DISABLE_COMP,                   Monitor disable completion
                                                           event
                                                           (Parameter: monitor_disale)
    RBLE_PXP_EVENT_MONITOR_ERROR_IND,                       Monitor error indication event
                                                           (Parameter: monitor_error_ind)
    RBLE_PXP_EVENT_MONITOR_READ_CHAR_RESPONSE,              Characteristic value read
                                                           request response event
                                                           (Parameter: rd_char_resp)
    RBLE_PXP_EVENT_MONITOR_WRITE_CHAR_RESPONSE,             Characteristic value write
                                                           request response event
                                                           (Parameter: wr_char_resp)
    RBLE_PXP_EVENT_MONITOR_COMMAND_DISALLOWED_IND,          Monitor command disallowed
                                                           indication event
                                                           (Parameter: cmd_disallowed_ind)
};
```

- Declaration of data type for PXP event types

```
typedef uint8_t  RBLE_PXP_EVENT_TYPE;
```

- Declaration of data type for PXP Reporter event callback function

```
typedef void ( *RBLE_PXPR_EVENT_HANDLER )( RBLE_PXPR_EVENT *event );
```

- Declaration of data type for PXP Monitor event callback function

```
typedef void ( *RBLE_PXPM_EVENT_HANDLER )( RBLE_PXPM_EVENT *event );
```

- Declaration of enumerated type for PXP Monitor alert service

```
enum RBLE_ALT_CODE_enum {
    RBLE_PROXM_SET_LK_LOSS_ALERT = 0x00,          Link loss service
    RBLE_PROXM_SET_IMMDT_ALERT                Immediate alert service
};
```

- Alert level characteristic structure

```
typedef struct RBLE_ALERT_LVL_CHAR_t {
    uint16_t      char_hdl;          Alert level characteristic handle
    uint16_t      val_hdl;          Alert level characteristic value handle
    uint8_t       prop;             Alert level characteristic property
    uint8_t       value;            Alert level value
}RBLE_ALERT_LVL_CHAR;
```

- Link loss service content structures

```
typedef struct RBLE_LLS_CONTENT_t {
    uint16_t      shdl;              Link loss service start handle
    uint16_t      ehdl;              Link loss service end handle
    RBLE_ALERT_LVL_CHAR  alert_lvl;  Link loss service alert level
                                     characteristic
}RBLE_LLS_CONTENT;
```

- Immediate alert service content structures

```
typedef struct RBLE_PXP_IAS_CONTENT_t {
    uint16_t      shdl;              IAS start handle
    uint16_t      ehdl;              IAS end handle
    RBLE_ALERT_LVL_CHAR  alert_lvl;  IAS alert level characteristic
}RBLE_PXP_IAS_CONTENT;
```

- Tx power service content structures

```
typedef struct RBLE_PXP_IAS_CONTENT_t {
    uint16_t      shdl;              TPS start handle
    uint16_t      ehdl;              TPS end handle
    uint16_t      txpw_lvl_char_hdl; Tx power characteristic handle
    uint16_t      txpw_lvl_val_hdl;  Tx power level value handle
    uint16_t      txpw_lvl_cfg_hdl;  Tx power level configuration
                                     characteristic handle
    uint8_t       txpw_lvl_prop;      Tx power level property
    uint8_t       txpw_lvl;           Tx power level value
}RBLE_TPS_CONTENT;
```

- PXP Monitor role parameter structures

```
typedef struct RBLE_PROXI_MON_PARAM_t {  
    RBLE_LLS_CONTENT    lls;                Link loss service content  
    RBLE_PXP_IAS_CONTENT ias;                Immediate alert service content  
    RBLE_TPS_CONTENT    tps;                Tx power service content  
}RBLE_PROXI_MON_PARAM;
```

- Attribute data structures

```
typedef struct RBLE_ATT_INFO_DATA_t {  
    uint8_t    each_len;                Length of each element  
    uint8_t    len;                    Attribute data length  
    uint8_t    data[RBLE_ATT_MAX_VALUE]; Attribute data  
}RBLE_ATT_INFO_DATA;
```


- PXP Reporter event parameter structures

```
typedef struct RBLE_PXPR_EVENT_t {
    RBLE_PXP_EVENT_TYPE      type;                PXP event type
    uint8_t                  reserved;             Reserved
    union Event_Pmr_Parameter_u {
        Generic event
        RBLE_STATUS          status;              Status

        Reporter enable completion event
        struct RBLE_PXP_Reporter_Enable_t{
            RBLE_STATUS      status;              Status
            uint8_t          reserved;            Reserved
            uint16_t         conhdl;              Connection handle
        }report_enable;

        Reporter disable completion event
        struct RBLE_PXP_Reporter_Disable_t{
            uint16_t          conhdl;              Connection handle
            uint8_t           lls_alert_lvl;       Link loss service alert level
            uint8_t           reserved;            Reserved
        }report_disable;

        Reporter alert indication event
        struct RBLE_PXP_Reporter_Alert_Ind_t{
            uint16_t          conhdl;              Connection handle
            uint8_t           alert_lvl;           Alert level
            uint8_t           reserved;            Reserved
        }report_alert_ind;

        Reporter command disallowed indication event
        struct RBLE_PXP_Reporter_Command_Disallowed_Ind_t{
            RBLE_STATUS      status;              Status
            uint8_t          reserved;            Reserved
            uint16_t         opcode;              Opcode
        }cmd_disallowed_ind;
    } param;
} RBLE_PXPR_EVENT;
```

- PXP Monitor event parameter structures

```
typedef struct RBLE_PXPM_EVENT_t {
    RBLE_PXP_EVENT_TYPE      type;           PXP event type
    uint8_t                  reserved;       Reserved
    union Event_Pmm_Parameter_u {
        Generic event
        RBLE_STATUS          status;        Status

        Monitor enable completion event
        struct RBLE_PXP_Monitor_Enable_t{
            RBLE_STATUS      status;        Status
            uint8_t          reserved;      Reserved
            uint16_t         conhdl;       Connection handle
            RBLE_LLS_CONTENT lls;          Link loss service information
            RBLE_PXP_IAS_CONTENT ias;      Immediate alert service
                                           information
            RBLE_TPS_CONTENT tps;          Tx power service information
        }monitor_enable;

        Monitor disable completion event
        struct RBLE_PXP_Monitor_Disable_t{
            RBLE_STATUS      status;        Status
            uint8_t          reserved;      Reserved
            uint16_t         conhdl;       Connection handle
        }monitor_disale;

        Monitor error indication event
        struct RBLE_PXP_Monitor_Error_Ind_t{
            RBLE_STATUS      status;        Status
            uint8_t          reserved;      Reserved
            uint16_t         conhdl;       Connection handle
        }monitor_error_ind;

        Monitor characteristic value read request response event
        struct RBLE_PXP_Monitor_Read_Char_Response_t{
            uint16_t         conhdl;       Connection handle
            uint8_t          att_code;     Status
            RBLE_ATT_INFO_DATA data;       Read characteristic data
        }rd_char_resp;

        Monitor characteristic value write request response event
        struct RBLE_PXP_Monitor_Write_Char_Response_t{
            uint16_t         conhdl;       Connection handle
            uint8_t          att_code;     Status
        }wr_char_resp;
    };
};
```

Monitor command disallowed indication event

```
struct RBLE_PXP_Monitor_Command_Disallowed_Ind_t{
    RBLE_STATUS      status;      Status
    uint8_t          reserved;    Reserved
    uint16_t          opcode;     Opcode
}cmd_disallowed_ind;
} param;
} RBLE_PXPM_EVENT;
```

3.2 Functions

The following table shows the API functions defined for the PXP of rBLE and the following sections describe the API functions in detail.

Table 3-1 API Functions Used by the PXP

RBLE_PXP_Reporter_Enable	Enables the Reporter role.
RBLE_PXP_Reporter_Disable	Disables the Reporter role.
RBLE_PXP_Monitor_Enable	Enables the Monitor role.
RBLE_PXP_Monitor_Disable	Disables the Monitor role.
RBLE_PXP_Monitor_Get_Alert_Level	Acquires the alert level value.
RBLE_PXP_Monitor_Set_Alert_Level	Specifies the alert level value.
RBLE_PXP_Monitor_Get_Tx_Power	Acquires the Tx power.

3.2.1 RBLE_PXP_Reporter_Enable

RBLE_STATUS RBLE_PXP_Reporter_Enable(uint16_t conhdl, uint8_t alert_lvl, uint8_t sec_lvl, RBLE_PXPR_EVENT_HANDLER call_back)

This function enables the PXP Reporter role.

The result is reported by using the Reporter role enable completion event RBLE_PXP_EVENT_REPORTER_ENABLE_COMP.

Parameters:

<i>conhdl</i>	Connection handle		
<i>alert_lvl</i>	RBLE_SVC_ALERT_NONE	No alert	
	RBLE_SVC_ALERT_MILD	Mild alert	
	RBLE_SVC_ALERT_HIGH	High alert	
<i>sec_lvl</i>	Security level		
<i>call_back</i>	Specify the callback function that reports the PXP event.		

Return:

<i>RBLE_OK</i>	Success
<i>RBLE_ERR</i>	Failed to allocate the area for the callback function.
<i>RBLE_PARAM_ERR</i>	Invalid parameter
<i>RBLE_STATUS_ERROR</i>	Not executable because the rBLE mode is other than RBLE_MODE_ACTIVE.

3.2.2 RBLE_PXP_Reporter_Disable

RBLE_STATUS RBLE_PXP_Reporter_Disable(uint16_t conhdl)

This function disables the PXP Reporter role.

The result is reported by using the Reporter role disable completion event RBLE_PXP_EVENT_REPORTER_DISABLE_COMP.

Parameters:

<i>conhdl</i>	Connection handle
---------------	-------------------

Return:

<i>RBLE_OK</i>	Success
<i>RBLE_STATUS_ERROR</i>	Not executable because the rBLE mode is other than RBLE_MODE_ACTIVE.

3.2.3 RBLE_PXP_Monitor_Enable

```
RBLE_STATUS RBLE_PXP_Monitor_Enable(uint16_t conhdl, uint8_t con_type,
    RBLE_PROXI_MON_PARAM *param, RBLE_PXPM_EVENT_HANDLER call_back)
```

This function enables the Proximity Monitor role and starts access to the service exposed by the Proximity Reporter. The result is reported by using the Monitor role enable completion event RBLE_PXP_EVENT_MONITOR_ENABLE_COMP.

When starting access to the service exposed by a Proximity Reporter to be connected for the first time, set 0 to the parameter of the service to configure the connection and to discover the service for the Reporter. If the handle information about the discovered service is saved and is used when the Reporter is connected normally for a second or subsequent time, detecting the service is skipped, which enables a high-speed access to the service.

While the Monitor role is enabled, the service exposed by only one Proximity Reporter is accessible. To connect to more than one Reporter at the same time and access the service exposed by each Reporter, repeat enable/disable of the Proximity Monitor role in order to switch access to them. At that time, perform normal connection by using the connection handle (which was obtained when connecting to each Reporter) and the handle information (which was saved when starting access to the service for the first time) as parameters.

Parameters:

<i>conhdl</i>	Connection handle			
<i>con_type</i>	RBLE_PRF_CON_DISCOVERY			Configuration connection performed when connecting for the first time
	RBLE_PRF_CON_NORMAL			Normal connection performed when connecting for the second and subsequent times
<i>*param</i>	<i>lls</i>	<i>shdl</i>		Link loss service start handle
		<i>ehdl</i>		Link loss service end handle
		<i>alert_lvl</i>	<i>char_hdl</i>	Alert level characteristic handle
			<i>val_hdl</i>	Alert level characteristic value handle
			<i>prop</i>	Alert level characteristic property
			<i>value</i>	Alert level value
	<i>ias</i>	<i>shdl</i>		Immediate alert service start handle
		<i>ehdl</i>		Immediate alert service end handle
		<i>alert_lvl</i>	<i>char_hd</i>	Alert level characteristic handle
			<i>val_hdl</i>	Alert level characteristic value handle
			<i>prop</i>	Alert level characteristic property
			<i>value</i>	Alert level value
	<i>tps</i>	<i>shdl</i>		Tx power service start handle
		<i>ehdl</i>		Tx power service end handle
		<i>txpw_lvl_char_hdl</i>		Tx power level characteristic handle
		<i>txpw_lvl_val_hdl</i>		Tx power level characteristic value handle
		<i>txpw_lvl_cfg_hdl</i>		Tx power level client characteristic configuration descriptor property
		<i>txpw_lvl_prop</i>		Tx power level characteristic property
		<i>txpw_lvl</i>		Tx power level value
<i>call_back</i>	Specify the callback function that reports the PXP event.			

Return:

<i>RBLE_OK</i>	Success
<i>RBLE_ERR</i>	Failed to allocate the area for the callback function.
<i>RBLE_PARAM_ERR</i>	Invalid parameter
<i>RBLE_STATUS_ERROR</i>	Not executable because the rBLE mode is other than RBLE_MODE_ACTIVE.

3.2.4 RBLE_PXP_Monitor_Disable

RBLE_STATUS RBLE_PXP_Monitor_Disable(uint16_t conhdl)	
This function disables the Proximity Monitor role and terminate access to the service exposed by the Proximity Reporter.	
The result is reported by using the Monitor role disable completion event RBLE_PXP_EVENT_MONITOR_DISABLE_COMP.	
Parameters:	
<i>conhdl</i>	Connection handle
Return:	
<i>RBLE_OK</i>	Success
<i>RBLE_STATUS_ERROR</i>	Not executable because the rBLE mode is other than RBLE_MODE_ACTIVE.

3.2.5 RBLE_PXP_Monitor_Get_Alert_Level

RBLE_STATUS RBLE_PXP_Monitor_Get_Alert_Level(uint16_t conhdl)	
This function acquires the alert level value from the link loss service alert level characteristic.	
Parameters:	
<i>conhdl</i>	Connection handle
Return:	
<i>RBLE_OK</i>	Success
<i>RBLE_STATUS_ERROR</i>	Not executable because the rBLE mode is other than RBLE_MODE_ACTIVE.

3.2.6 RBLE_PXP_Monitor_Set_Alert_Level

RBLE_STATUS RBLE_PXP_Monitor_Set_Alert_Level(uint16_t conhdl,uint8_t svc_code,uint8_t lvl)			
This function sets the alert level value to the alert level characteristic of the link loss service or immediate alert service.			
Parameters:			
conhdl	Connection handle		
svc_code	RBLE_PROXM_SET_LK_LOSS_ALERT	Link loss service	
	RBLE_PROXM_SET_IMMDT_ALERT	Immediate alert service	
lvl	RBLE_SVC_ALERT_NONE	No alert	
	RBLE_SVC_ALERT_MILD	Mild alert	
	RBLE_SVC_ALERT_HIGH	High alert	
Return:			
RBLE_OK	Success		
RBLE_STATUS_ERROR	Not executable because the rBLE mode is other than RBLE_MODE_ACTIVE.		

3.2.7 R_BLE_PXP_Monitor_Get_Tx_Power

RBLE_STATUS RBLE_PXP_Monitor_Get_Tx_Power(uint16_t conhdl)	
This function acquires the Tx power level value from the Tx power service Tx power level characteristic.	
Parameters:	
<i>conhdl</i>	Connection handle
Return:	
<i>RBLE_OK</i>	Success
<i>RBLE_STATUS_ERROR</i>	Not executable because the rBLE mode is other than RBLE_MODE_ACTIVE.

3.3 Events

The following table shows the events defined for the PXP of rBLE and the following sections describe the events in detail.

Table 3-2 Events Defined for the PXP

RBLE_PXP_EVENT_REPORTER_ENABLE_COMP	Reporter role enable completion event
RBLE_PXP_EVENT_REPORTER_DISABLE_COMP	Reporter role disable completion event
RBLE_PXP_EVENT_REPORTER_ALERT_IND	Reporter alert indication event
RBLE_PXP_EVENT_REPORTER_COMMAND_DISALLOWED_IND	Reporter role command disallowed indication event
RBLE_PXP_EVENT_MONITOR_ENABLE_COMP	Monitor role enable completion event
RBLE_PXP_EVENT_MONITOR_DISABLE_COMP	Monitor role disable completion event
RBLE_PXP_EVENT_MONITOR_ERROR_IND	Monitor role error indication event
RBLE_PXP_EVENT_MONITOR_READ_CHAR_RESPONSE	Characteristic value read request response event
RBLE_PXP_EVENT_MONITOR_WRITE_CHAR_RESPONSE	Characteristic value write request response event
RBLE_PXP_EVENT_MONITOR_COMMAND_DISALLOWED_IND	Monitor role command disallowed indication event

3.3.1 RBLE_PXP_EVENT_REPORTER_ENABLE_COMP

RBLE_PXP_EVENT_REPORTER_ENABLE_COMP	
This event reports the result of enabling the Reporter role (RBLE_PXP_Reporter_Role_Enable).	
Parameters:	
<i>status</i>	Result of enabling the Reporter role (See 2.2 and <i>Bluetooth Low Energy Protocol Stack API Reference Manual: Basics, 3.2, Declaration of enumerated type for rBLE status.</i>)
<i>conhdl</i>	Connection handle

3.3.2 RBLE_PXP_EVENT_REPORTER_DISABLE_COMP

RBLE_PXP_EVENT_REPORTER_DISABLE_COMP			
This event reports the result of disabling the Reporter role (RBLE_PXP_Reporter_Role_Disable).			
Parameters:			
<i>conhdl</i>	Connection handle		
<i>lls_alert_lvl</i>	RBLE_SVC_ALERT_NONE	No alert	
	RBLE_SVC_ALERT_MILD	Mild alert	
	RBLE_SVC_ALERT_HIGH	High alert	

3.3.3 RBLE_PXP_EVENT_REPORTER_ALERT_IND

RBLE_PXP_EVENT_REPORTER_ALERT_IND			
This event indicates the Reporter alert level. This event indicates the immediate alert service alert value received from the Monitor and the alert value of the link loss service when the link was lost.			
Parameters:			
<i>conhdl</i>	Connection handle		
<i>alert_lvl</i>	RBLE_SVC_ALERT_NONE	No alert	
	RBLE_SVC_ALERT_MILD	Mild alert	
	RBLE_SVC_ALERT_HIGH	High alert	

3.3.4 RBLE_PXP_EVENT_REPORTER_COMMAND_DISALLOWED_IND

RBLE_PXP_EVENT_REPORTER_COMMAND_DISALLOWED_IND			
This event indicates the error that occurs when a command executed by the Reporter role cannot be accepted.			
Parameters:			
<i>status</i>	Result of command execution (See 2.2 and <i>Bluetooth Low Energy Protocol Stack API Reference Manual: Basics, 3.2, Declaration of enumerated type for rBLE status.</i>)		
<i>opcode</i>	RBLE_CMD_PXP_REPORTER_ENABLE	Reporter role enable command	
	RBLE_CMD_PXP_REPORTER_DISABLE	Reporter role disable command	

3.3.5 RBLE_PXP_EVENT_MONITOR_ENABLE_COMP

RBLE_PXP_EVENT_MONITOR_ENABLE_COMP

This event reports the result of enabling the Monitor role (RBLE_PXP_Monitor_Role_Enable).
Save the obtained handle information about the discovered service, to enable a high-speed access to the service without service detection when restarting access to the service.

Parameters:

<i>status</i>	Result of enabling the Monitor role (See 2.2 and Bluetooth Low Energy Protocol Stack API Reference Manual: Basics, 3.2, Declaration of enumerated type for rBLE status.)		
<i>conhdl</i>	Connection handle		
<i>lls</i>	<i>shdl</i>		Link loss service start handle
	<i>ehdl</i>		Link loss service end handle
	<i>alert_lvl</i>	<i>char_hdl</i>	Alert level characteristic handle
		<i>val_hdl</i>	Alert level characteristic value handle
		<i>prop</i>	Alert level characteristic property
		<i>value</i>	Alert level value
<i>ias</i>	<i>shdl</i>		Immediate alert service start handle
	<i>ehdl</i>		Immediate alert service end handle
	<i>alert_lvl</i>	<i>char_hdl</i>	Alert level characteristic handle
		<i>val_hdl</i>	Alert level characteristic value handle
		<i>prop</i>	Alert level characteristic property
		<i>value</i>	Alert level value
<i>tps</i>	<i>shdl</i>		Tx power service start handle
	<i>ehdl</i>		Tx power service end handle
	<i>txpw_lvl_char_hdl</i>		Tx power level characteristic handle
	<i>txpw_lvl_val_hdl</i>		Tx power level characteristic value handle
	<i>txpw_lvl_cfg_hdl</i>		Tx power level client characteristic configuration descriptor property
	<i>txpw_lvl_prop</i>		Tx power level characteristic property
	<i>txpw_lvl</i>		Tx power level value

3.3.6 RBLE_PXP_EVENT_MONITOR_DISABLE_COMP

RBLE_PXP_EVENT_MONITOR_DISABLE_COMP

This event reports the result of disabling the Monitor role (RBLE_PXP_Monitor_Role_Disable).

Parameters:

<i>status</i>	Result of disabling the Monitor role (See 2.2 and Bluetooth Low Energy Protocol Stack API Reference Manual: Basics, 3.2, Declaration of enumerated type for rBLE status.)
<i>conhdl</i>	Connection handle

3.3.7 RBLE_PXP_EVENT_MONITOR_ERROR_IND

RBLE_PXP_EVENT_MONITOR_ERROR_IND		
This event indicates an error code unique to the Monitor role.		
Parameters:		
<i>status</i>	Error code (See 2.2 and <i>Bluetooth Low Energy Protocol Stack API Reference Manual: Basics, 3.2, Declaration of enumerated type for rBLE status.</i>)	
<i>conhdl</i>	Connection handle	

3.3.8 RBLE_PXP_EVENT_MONITOR_READ_CHAR_RESPONSE

RBLE_PXP_EVENT_MONITOR_READ_CHAR_RESPONSE			
This event reports the response to the characteristic value acquisition request (RBLE_PXP_Monitor_Get_Alert_Level, RBLE_PXP_Monitor_Get_Tx_Power). Read out the acquired data in accordance with the contents of the request.			
Parameters:			
<i>conhdl</i>	Connection handle		
<i>att_code</i>	0x00	Characteristic value successfully acquired	
	Other than 0x00	Error occurred when acquiring characteristic value	
<i>data</i>	<i>each_len</i>	Length of each result	
	<i>len</i>	Data length	
	<i>data</i> [RBLE_ATT_M_MAX_VALUE]	Read characteristic data	

3.3.9 RBLE_PXP_EVENT_MONITOR_WRITE_CHAR_RESPONSE

RBLE_PXP_EVENT_MONITOR_WRITE_CHAR_RESPONSE			
This event reports the response to the characteristic value setting request (RBLE_PXP_Monitor_Set_Alert_Level).			
Parameters:			
<i>conhdl</i>	Connection handle		
<i>att_code</i>	0x00	Characteristic value successfully set	
	Other than 0x00	Error occurred when setting characteristic value	

3.3.10 RBLE_PXP_EVENT_MONITOR_COMMAND_DISALLOWED_IND

RBLE_PXP_EVENT_MONITOR_COMMAND_DISALLOWED_IND			
This event indicates the error that occurs when a command executed by the Monitor role cannot be accepted.			
Parameters:			
<i>status</i>	Result of command execution (See 2.2 and <i>Bluetooth Low Energy Protocol Stack API Reference Manual: Basics, 3.2, Declaration of enumerated type for rBLE status.</i>)		
<i>opcode</i>	RBLE_CMD_PXP_MONITOR_ENABLE		Monitor role enable command
	RBLE_CMD_PXP_MONITOR_DISABLE		Monitor role disable command
	RBLE_CMD_PXP_MONITOR_GET_ALERT_LEVEL		Alert level acquisition command
	RBLE_CMD_PXP_MONITOR_SET_ALERT_LEVEL		Alert level setting command
	RBLE_CMD_PXP_MONITOR_GET_TX_POWER		Tx power acquisition command

3.4 Message Sequence Chart

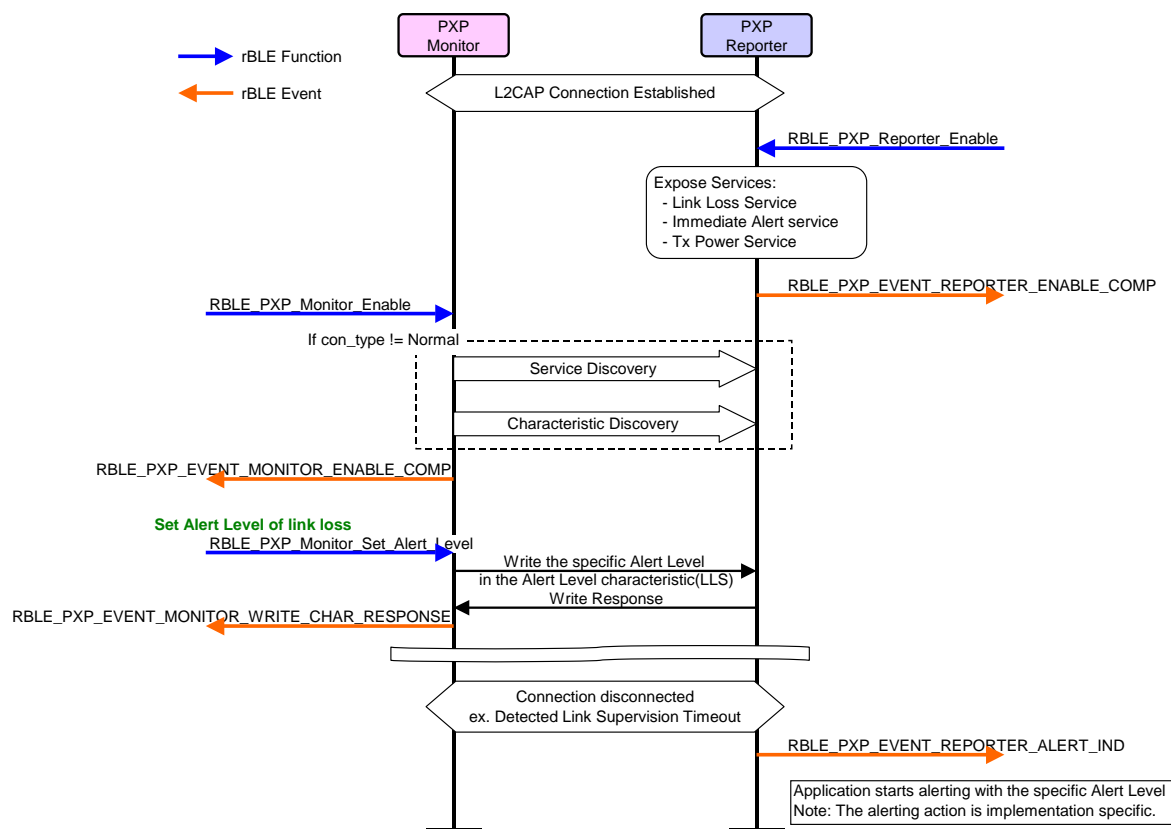


Figure 3-1 Example of Use Case In Which PXP Is Implemented by Using rBLE API (Using LLS)

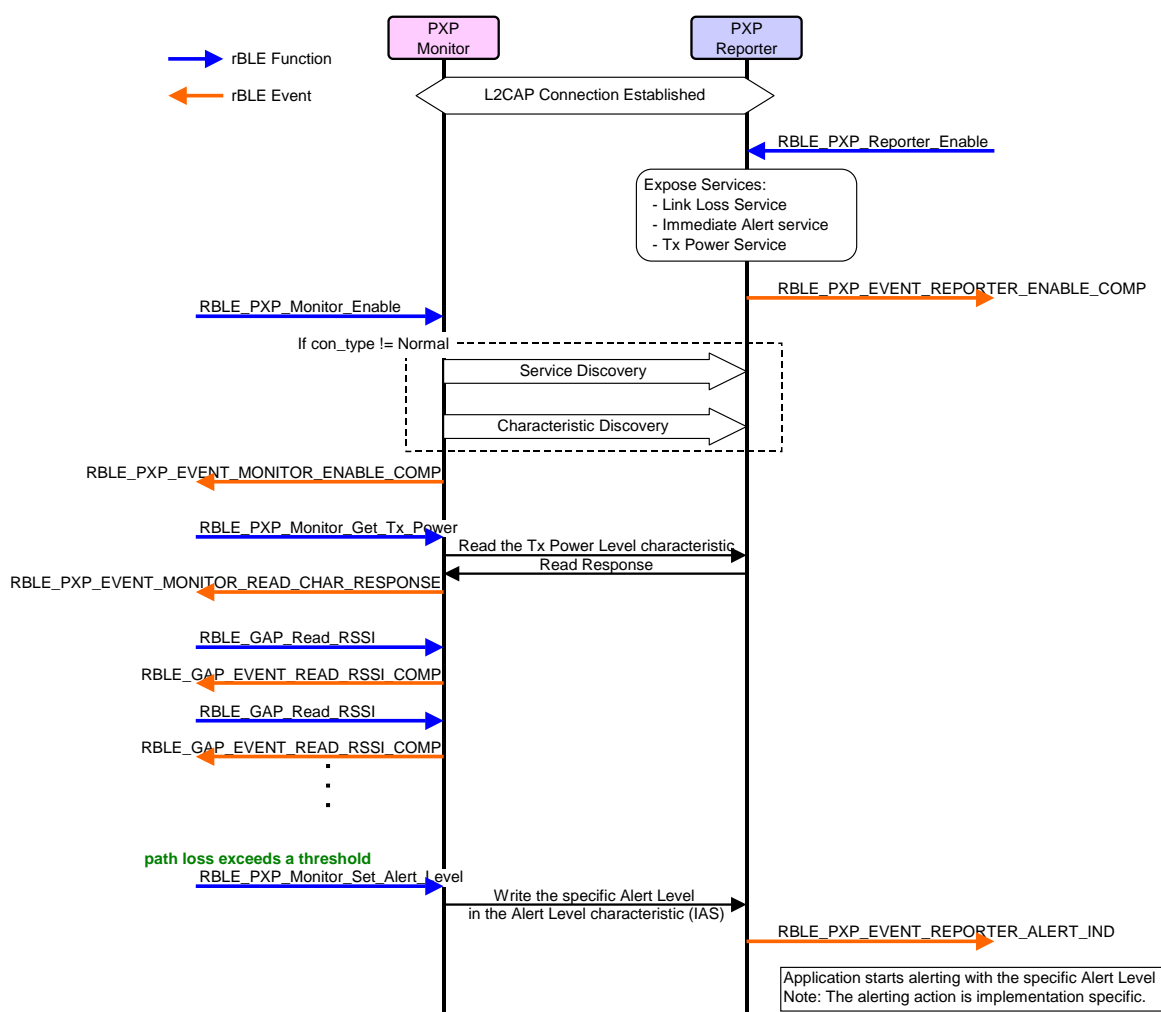


Figure 3-2 Example of Use Case In Which PXP Is Implemented by Using rBLE API (Using IAS)

4. Notes

Appendix A How to Read Definition Tables

This section shows how to read the tables that describes the rBLE API functions and events shown in this document.

A.1 How to Read Function Definition Tables

The following contents are included in the function definition tables:

The Parameters area describes the parameters specified for the function. The italicized character strings on the left are the parameters of the function. The meaning of each parameter is described on the far right following the variables.

The italicized character string(s) next to each parameter indicate the member(s) of the parameter (structure).

The values that can be specified for the parameter might be described between the parameter name and its description.

The function definition is shown at the top of the table in the row with the light green background. This area shows the function prototype.

The operation of the function and the event reported after executing the function are described in this area.

Parameters:			
<i>Parameter 1</i>	Description of parameter 1		
<i>Parameter 2</i>	<i>Member 1</i>	Value 1 that can be specified for member 1	Description of value 1 that can be specified for member 1
		Value 1 that can be specified for member 2	Description of value 1 that can be specified for member 2
	<i>Member 2</i>	Description of member 2	
Return:			
<i>Value 1 that might be returned</i>		Description of value 1 that might be returned	
<i>Value 2 that might be returned</i>		Description of value 2 that might be returned	

The Return area describes the values returned for the function. The leftmost row shows the value that might be returned, and the next row describes the return value.

A.2 How to Read Event Definition Tables

The following contents are included in the event definition tables:

The Parameters area describes the parameters specified for the event. The italicized character strings on the left show the parameters of the event parameter structure. The meaning of each parameter is described on the far right.

The italicized character string(s) next to each parameter indicate the member(s) of the parameter (structure).

The event definition is shown at the top of the table in the row with the orange background. This area shows the event type.

The information reported by the event is described in this area.

Parameters:

<i>Parameter 1</i>	Description of parameter 1	
<i>Parameter 2</i>	<i>Member 1</i>	Description of member 1
	<i>Member 2</i>	Description of member 2
	<i>Member 3</i>	Description of member 3
<i>Parameter 3</i>	Value 1 that can be specified for parameter 3	Description of value 1 that can be specified for parameter 3
	Value 2 that can be specified for parameter 3	Description of value 2 that can be specified for parameter 3

The values that can be specified for the parameter might be shown between the parameter name and its description.

Appendix B Referenced Documents

1. Bluetooth Core Specification v4.0, Bluetooth SIG
2. Find Me Profile Specification v1.0, Bluetooth SIG
3. Immediate Alert Service Specification v1.0, Bluetooth SIG
4. Proximity Profile Specification v1.0, Bluetooth SIG
5. Link Loss Service Specification v1.0, Bluetooth SIG
6. Tx Power Service Specification v1.0, Bluetooth SIG
7. Health Thermometer Profile Specification v1.0, Bluetooth SIG
8. Health Thermometer Service Specification v1.0, Bluetooth SIG
9. Device Information Service Specification v1.1, Bluetooth SIG
10. Blood Pressure Profile Specification v1.0, Bluetooth SIG
11. Blood Pressure Service Specification v1.0, Bluetooth SIG
12. HID over GATT Profile Specification v1.0, Bluetooth SIG
13. HID Service Specification v1.0, Bluetooth SIG
14. Battery Service Specification v1.0, Bluetooth SIG
15. Scan Parameters Profile Specification v1.0, Bluetooth SIG
16. Scan Parameters Service Specification v1.0, Bluetooth SIG
17. Bluetooth SIG Assigned Numbers <https://www.bluetooth.org/Technical/AssignedNumbers/home.htm>
18. Services & Characteristics UUID <http://developer.bluetooth.org/gatt/Pages/default.aspx>
19. Personal Health Devices Transcoding White Paper v1.2, Bluetooth SIG

Appendix C Terminology

Term	Description
Service	A service is provided from a GATT server to a GATT client. The GATT server exposes some characteristics as the interface. The service prescribes how to access the exposed characteristics.
Profile	A profile enables implementation of a use case by using one or more services. The services used are defined in the specifications of each profile.
Characteristic	A characteristic is a value used to identify services. The characteristics to be exposed and their formats are defined by each service.
Role	Each device takes the role prescribed by the profile or service in order to implement the specified use case.
Client Characteristic Configuration Descriptor	A descriptor is used to control notifications or indications of characteristic values that include the client characteristic configuration descriptor sent from the GATT server.
Connection Handle	The handle determined by the controller stack and is used to identify connection with a remote device. The valid handle range is between 0x0000 and 0x0EFF.

REVISION HISTORY	Bluetooth Low Energy Protocol Stack API Reference Manual: PXP
------------------	---

Rev.	Date	Description	
		Page	Summary
1.00	Feb 15, 2013	---	First Edition issued
1.01	Mar 27, 2013	---	The description about the high-speed access to the service for a second or subsequent time is added.
1.02	Jun 28, 2013	---	Bookmark is added.
1.03	Sep 19, 2014	2	The common definitions of profile are added.
		5	A definition of connection type is deleted.
		---	Parameter description is changed to use the common definitions of profile.
1.04	Apr 17, 2015	2	The service definitions are updated.

Bluetooth Low Energy Protocol Stack
API Reference Manual: PXP

Publication Date: Rev.1.04 Apr 17, 2015

Published by: Renesas Electronics Corporation



SALES OFFICES

Renesas Electronics Corporation

<http://www.renesas.com>

Refer to "<http://www.renesas.com/>" for the latest and detailed information.

Renesas Electronics America Inc.

2801 Scott Boulevard Santa Clara, CA 95050-2549, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited

9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004

Renesas Electronics Europe Limited

Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K.
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH

Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.

Room 1709, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100191, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.

Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited

Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

Renesas Electronics Taiwan Co., Ltd.

13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.

80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.

Unit 1207, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics India Pvt. Ltd.

No.777C, 100 Feet Road, HALII Stage, Indiranagar, Bangalore, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.

12F., 234 Teheran-ro, Gangnam-Gu, Seoul, 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Bluetooth Low Energy Protocol Stack



Renesas Electronics Corporation

R01UW0090EJ0104