**Microchip-Logo with R.emfAudio UART Command Set (V2.02)**

Contents

[1 General description 7](#_Toc486513729)

[2 MCU interface 7](#_Toc486513730)

[2.1 PIN definition 7](#_Toc486513731)

[2.2 UART Protocol 7](#_Toc486513732)

[2.3 UART configuration 8](#_Toc486513733)

[2.4 UART data exchange for low power mode 8](#_Toc486513734)

[2.5 UART packet process rule 9](#_Toc486513735)

[2.5.1 Command Packet Handling 9](#_Toc486513736)

[2.5.2 Event Packet Handling 10](#_Toc486513737)

[2.5.3 Error handling in UART command 10](#_Toc486513738)

[3 Command/ EVENT OP code definition 11](#_Toc486513739)

[3.1 UART Commands 14](#_Toc486513740)

[3.1.1 Make\_Call (0x00) 14](#_Toc486513741)

[3.1.2 Make\_Extension\_Call (0x01) 16](#_Toc486513742)

[3.1.3 Event\_Filter\_Setting (0x03) 17](#_Toc486513743)

[3.1.4 Music\_Control (0x04) 18](#_Toc486513744)

[3.1.5 Change\_Device\_Name (0x05) 19](#_Toc486513745)

[3.1.6 Change\_PIN\_Code (0x06) 19](#_Toc486513746)

[3.1.7 BTM\_Parameter\_Setting (0x07) 20](#_Toc486513747)

[3.1.8 Read\_BTM\_Version (0x08) 22](#_Toc486513748)

[3.1.9 Vendor\_AT\_Cmd (0x0A) 22](#_Toc486513749)

[3.1.10 AVC\_Vendor\_Dependent\_Cmd (0x0B) 23](#_Toc486513750)

[3.1.11 AVC\_Group\_Navigation (0x0C) 24](#_Toc486513751)

[3.1.12 Read\_Link\_Status (0x0D) 25](#_Toc486513752)

[3.1.13 Read\_Paired\_Device\_Record (0x0E) 25](#_Toc486513753)

[3.1.14 Read\_Local\_BD\_Address (0x0F) 26](#_Toc486513754)

[3.1.15 Read\_Local\_Device\_Name (0x10) 26](#_Toc486513755)

[3.1.16 Send\_SPP/iAP\_Or\_LE\_Data (0x12) 27](#_Toc486513756)

[3.1.17 BTM\_Utility\_Function (0x13) 28](#_Toc486513757)

[3.1.18 Event\_Ack (0x14) 33](#_Toc486513758)

[3.1.19 Additional\_Profile\_Link\_Setup (0x15) 33](#_Toc486513759)

[3.1.20 Read\_Linked\_Device\_Information (0x16) 34](#_Toc486513760)

[3.1.21 Profile\_Link\_Back (0x17) 35](#_Toc486513761)

[3.1.22 Disconnect (0x18) 37](#_Toc486513762)

[3.1.23 User\_Confirm\_SSP\_Req\_Reply (0x1A) 37](#_Toc486513763)

[3.1.24 Set\_HF\_Speaker\_Gain\_Level (0x1B) 38](#_Toc486513764)

[3.1.25 EQ\_Mode\_Setting (0x1C) 39](#_Toc486513765)

[3.1.26 DSP\_NR\_CTRL (0x1D) 39](#_Toc486513766)

[3.1.27 GPIO\_CTRL (0x1E) 40](#_Toc486513767)

[3.1.28 MCU\_UART\_Rx\_Buffer\_Size (0x1F) 43](#_Toc486513768)

[3.1.29 Voice\_Prompt\_Cmd (0x20) 43](#_Toc486513769)

[3.1.30 Set\_Overall\_Gain (0x23) 45](#_Toc486513770)

[3.1.31 Read\_BTM\_Setting (0x24) 46](#_Toc486513771)

[3.1.32 Read\_BTM\_Battery\_Charger\_Status (0x25) 47](#_Toc486513772)

[3.1.33 MCU\_Update\_Cmd (0x26) 48](#_Toc486513773)

[3.1.34 Report\_Battery\_Capacity (0x27) 48](#_Toc486513774)

[3.1.35 LE\_ANCS\_Service\_Cmd (0x28) 49](#_Toc486513775)

[3.1.36 LE\_Signaling\_Cmd (0x29) 50](#_Toc486513776)

[3.1.37 nSPK\_Vendor\_Cmd (0x2A) 52](#_Toc486513777)

[3.1.38 Read\_nSPK\_Link\_Status (0x2B) 53](#_Toc486513778)

[3.1.39 nSPK\_Sync\_Audio\_Effect (0x2C) 53](#_Toc486513779)

[3.1.40 LE\_GATT\_CMD (0x2D) 54](#_Toc486513780)

[3.1.41 LE\_App\_Cmd (0x2F) 57](#_Toc486513781)

[3.1.42 DSP\_Runtime\_Program (0x30) 59](#_Toc486513782)

[3.1.43 Read\_Vendor\_Eeprom\_Data (0x31) 60](#_Toc486513783)

[3.1.44 Read\_IC\_Version\_Info (0x32) 60](#_Toc486513784)

[3.1.45 Voice\_Prompt\_Ind\_Cmd (0x33) 61](#_Toc486513785)

[3.1.46 Read\_BTM\_Link\_Mode (0x34) 62](#_Toc486513786)

[3.1.47 Configure\_Vendor\_Parameter (0x35) 62](#_Toc486513787)

[3.1.48 nSPK Exchange\_Link\_Info\_Cmd (0x37) 63](#_Toc486513788)

[3.1.51 nSPK Set GIAC(0x38) 64](#_Toc486513839)

[3.1.52 READ\_FEATURE\_LIST (0x39) 64](#_Toc486513840)

[3.1.53 Personal\_MSPK\_GROUP\_Control (0x3A) 65](#_Toc486513841)

[3.1.54 UART\_CMD\_TEST\_DEVICE (0x3B) 65](#_Toc486513842)

[4 MMI\_Action UART Command (0x02) 66](#_Toc486513843)

[5 Events 98](#_Toc486513844)

[5.1 Command\_Ack (0x00) 98](#_Toc486513845)

[5.2 BTM\_Status (0x01) 98](#_Toc486513846)

[5.3 Call\_Status (0x02) 101](#_Toc486513847)

[5.4 Caller\_ID (0x03) 102](#_Toc486513848)

[5.5 SMS\_Received\_Indication (0x04) 102](#_Toc486513849)

[5.6 Missed\_Call\_Indication (0x05) 103](#_Toc486513850)

[5.7 Phone\_Max\_Battery\_Level (0x06) 103](#_Toc486513851)

[5.8 Phone\_Current\_Battery\_Level (0x07) 104](#_Toc486513852)

[5.9 Roaming\_Status (0x08) 104](#_Toc486513853)

[5.10 Phone\_Max\_Signal\_Strength\_Level (0x09) 105](#_Toc486513854)

[5.11 Phone\_Current\_Signal\_Strength\_Level (0x0A) 105](#_Toc486513855)

[5.12 Phone\_Service\_Status (0x0B) 106](#_Toc486513856)

[5.13 BTM\_Battery\_Status (0x0C) 106](#_Toc486513857)

[5.14 BTM\_Charging\_Status (0x0D) 107](#_Toc486513858)

[5.15 Reset\_To\_Default (0x0E) 108](#_Toc486513859)

[5.16 Report\_HF\_Gain\_Level (0x0F) 109](#_Toc486513860)

[5.17 EQ\_Mode\_Indication (0x10) 109](#_Toc486513861)

[5.18 Read\_Linked\_Device\_Information\_Reply (0x17) 110](#_Toc486513862)

[5.19 Read\_BTM\_Version\_Reply (0x18) 112](#_Toc486513863)

[5.20 Call\_List\_Report (0x19) 112](#_Toc486513864)

[5.21 AVC\_Vendor\_Dependent\_Response (0x1A) 113](#_Toc486513865)

[5.22 BTM\_Utility\_Req (0x1B) 114](#_Toc486513866)

[5.23 Vendor\_AT\_Cmd\_Rsp (0x1C) 117](#_Toc486513867)

[5.24 Report\_Vendor\_AT\_Event (0x1D) 117](#_Toc486513868)

[5.25 Read\_Link\_Status\_Reply (0x1E) 118](#_Toc486513869)

[5.26 Read\_Paired\_Device\_Record\_Reply (0x1F) 119](#_Toc486513870)

[5.27 Read\_Local\_BD\_Address\_Reply (0x20) 120](#_Toc486513871)

[5.28 Read\_Local\_Device\_Name\_Reply (0x21) 120](#_Toc486513872)

[5.29 Reprt\_SPP/iAP/LE\_Data (0x22) 121](#_Toc486513873)

[5.30 Reprt\_Link\_Back\_Status (0x23) 122](#_Toc486513874)

[5.31 Ringtone\_Status\_Indication (0x24) 122](#_Toc486513875)

[5.32 User\_Confirm\_SSP\_Req (0x25) 123](#_Toc486513876)

[5.33 Report\_AVRCP\_Volume\_Ctrl (0x26) 123](#_Toc486513877)

[5.34 Report\_Input\_Signal\_Level (0x27) 124](#_Toc486513878)

[5.35 Report\_iAP\_Info (0x28) 126](#_Toc486513879)

[5.36 Report\_AVRCP\_ABS\_Volume\_Level (0x29) 127](#_Toc486513880)

[5.37 Report\_Voice\_Prompt\_Status (0x2A) 127](#_Toc486513881)

[5.38 Report\_Type\_Codec (0x2D) 128](#_Toc486513882)

[5.39 Report\_Type\_BTM\_Settings (0x2E) 128](#_Toc486513883)

[5.40 Report\_MCU\_Update\_Reply (0x2F) 129](#_Toc486513884)

[5.41 Report\_BTM\_Initial\_Status (0x30) 130](#_Toc486513885)

[5.42 LE\_ANCS\_Service\_Event (0x31) 130](#_Toc486513886)

[5.43 LE\_Signaling\_Event (0x32) 132](#_Toc486513887)

[5.44 Report\_nSPK\_Link\_Status (0x33) 134](#_Toc486513888)

[5.45 Report\_nSPK\_Vendor\_Event (0x34) 135](#_Toc486513889)

[5.46 Report\_nSPK\_Audio\_Setting (0x35) 136](#_Toc486513890)

[5.47 Report\_Sound\_Effect\_Status (0x36) 137](#_Toc486513891)

[5.48 Report\_Vendor\_EEPROM\_Data (0x37) 137](#_Toc486513892)

[5.49 Report\_IC\_Ver\_Info (0x38) 138](#_Toc486513893)

[5.50 REPORT\_LE\_GATT\_EVENT (0x39) 139](#_Toc486513894)

[5.50.1 Client\_write\_char\_value (0x00) 139](#_Toc486513895)

[5.50.2 Read\_Local\_Char\_Value\_Res (0x01) 140](#_Toc486513896)

[5.50.3 Discover\_All\_Primary\_Services\_Res (0x02) 140](#_Toc486513897)

[5.50.4 Discover\_Specific\_Primary\_Service\_Characteristics\_Res (0x03) 141](#_Toc486513898)

[5.50.5 Discover\_All\_Characteristic\_Descriptors\_Res (0x04) 142](#_Toc486513899)

[5.50.6 Get\_Att\_MTU\_Size\_Res (0x05) 143](#_Toc486513900)

[5.51 Report\_BTM\_Link\_Mode (0x3A) 143](#_Toc486513901)

[5.52 DSP\_Dedicated\_Event (0x3B) 144](#_Toc486513902)

[5.53 Report\_nSPK\_MISC\_Event (0x3C) 145](#_Toc486513903)

[5.54 Report\_nSPK\_Exchange\_Link\_info (0x3D) 145](#_Toc486513904)

[5.55 Report\_Customized\_Information (0x3E) 146](#_Toc486513905)

[5.56 Report\_CSB\_CLK (0x3F) 146](#_Toc486513906)

[5.57 Report\_Read\_Feature\_List\_Reply (0x40) 147](#_Toc486513907)

[5.58 REPORT\_TEST\_RESULT\_REPLY (0x41) 148](#_Toc486513908)

[6 Revision history 148](#_Toc486513909)

[7 Remove MMI commands 149](#_Toc486513910)

[8 References 151](#_Toc486513911)

[9 Terms and Definitions 151](#_Toc486513912)

# General description

**Dual Mode SPK** provides UART communication interface with MCU. This document describes how MCU communicates with **Dual Mode SPK** and the behavior of **Dual Mode SPK**.

# MCU interface

## PIN definition

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Description | BM62  Define |
| UART\_TXD  (Mandatory) | Output |  | HCI\_TXD |
| UART\_RXD  (Mandatory) | Input |  | HCI\_RXD |
| UART\_TX\_IND  (Optional) | Output | **DSPK** inform Host MCU that UART data will be transmitted out after few ms (Setting by UI Tool, default 9.375ms) | P2\_7 |
| UART\_RX\_IND  (Optional) | Input | Host MCU inform **DSPK** that UART data will be transmitted out after few us | Configurable |

## UART Protocol

The UART protocol is shown as below diagram

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **HEAD** | | **MID** | **DATA** | **CRC** |
|  | **START** | **LENGTH** | **OP Code** | **PARAMETER** | **CHKSUM** |
| **BYTE NO** | 0 | 1 ~ 2 | 3 | 4 ~ XX | Length + 3 |
| **SIZE (BYTE)** | 1 | 2 | 1 | 0～ | 1 |
| **VALUE** | 0xAA | 1～ | Command/Event | Command/Event parameter | Check sum |
|  | SINC WORD | Check sum to be calculated | | |  |
|  |  |  | TARGET LENGTH | |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Check sum rule: Summation of every byte after START WORD(LENGTH, COM.ID, COM PARAM, CHK SUM) is 0xXX00 | | | | | | |
| e.g. |  |  |  |  |  |  |
|  | **START** | **LENGTH(H)** | **LENGTH(L)** | **OP CODE** | **PARAMETER** | **CHKSUM** |
| **BYTE NO** | 0 | 1 | 2 | 3 | 4 | 5 |
| **VALUE** | 0xAA | 0x00 | 0x02 | 0x01 | 0x00 | 0xFD |

There is one EEPROM option (0xAE @ bit 4) to add one byte “0x00” as wakeup byte in front of start byte (0xAA) in UART event. This option is on by default.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Wakeup** | **START** | **LENGTH(H)** | **LENGTH(L)** | **OP CODE** | **PARAMETER** | **CHKSUM** |
| **BYTE NO** | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
| **VALUE** | **0x00** | 0xAA | 0x00 | 0x02 | 0x01 | 0x00 | 0xFD |

## UART configuration

* Baud Rate Setting: 2400bps~921600bps
* No Flow Control

## UART data exchange for low power mode

1. Signal of UART\_TX\_IND and UART\_RX\_IND are required to guarantee the correction of UART data.

UART\_RX\_IND

UART\_RX Start

UART\_RX End

**Tuart\_rx\_ind**

\***Tuart\_rx\_ind**: >2ms

Fig 2.4.1 Host MCU indicate DSPK UART data diagram

UART\_TXDUART\_RXD

UART\_TX\_IND

UART\_TX Start

UART\_TX End

**TWakeup\_External\_MCU\_Wait\_Time**

\*TWakeup\_External\_MCU\_Wait\_Time: The time before UART TXD send (set by UI)

Default value of TWakeup\_External\_MCU\_Wait\_Time: is 9.375 msec

Fig 2.4.2 DSPK indicate Host MCU UART data timing diagram

## UART packet process rule

### Command Packet Handling

For every command received from MCU, DSPK will send the ACK. If ACK is not received by MCU with in the 200ms timeout value then MCU has to re send the same command. After re sending the command also if there is no ACK then MCU can reset the DSPK.

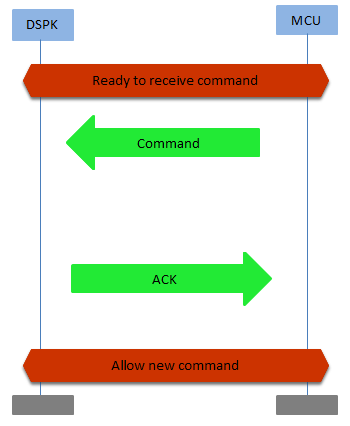
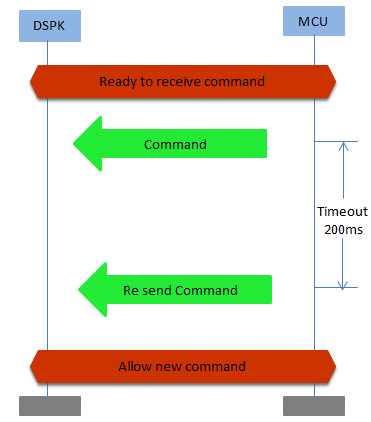
 

Fig 2.5.1: For the command received from Fig 2.5.2:After sending the command,MCU did

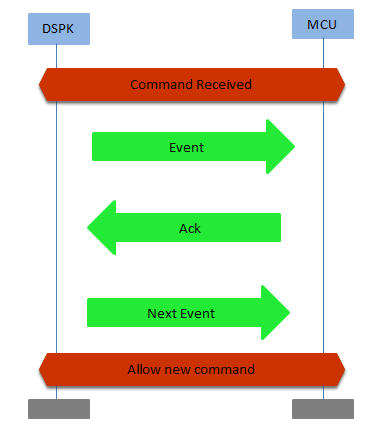
MCU,DSPK sent an ACK not receive the ACK within 200ms,So

MCU resending the same command

If the precondition is not met for any of the commands, then, the DSPK will send Command\_ACK Event packet with status = “command disallow”. Deviation from this behavior, if any, will be mentioned in the individual command description.

### Event Packet Handling

After sending Event from DSPK to MCU.DSPK will wait for 800 ms timeout period. If ACK is received from MCU within this time or timeout happened then the next event will be sent.



### Error handling in UART command

In different version of firmware design, the error handling might be different. The designer needs to follow the appropriate working scenario to use the UART command.

# Command/ EVENT OP code definition

#### UART Command

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| UART Command | |  | Support version | |
| OP code | Command | Return event | Start | End |
| 0x00 | [Make\_Call](#_Make_Call_(0x00)) | [Call\_Status](#_Call_Status_(0x02)) | V2.00 |  |
| 0x01 | [Make\_Extension\_Call](#_Make_Extension_Call_(0x01)) |  | V2.00 |  |
| 0x02 | [MMI\_Action](#_MMI_Action_UART_Command) |  | V2.00 |  |
| 0x03 | [Event\_Mask\_Setting](#_Event_Filter_Setting_(0x03)) |  | V2.00 |  |
| 0x04 | [Music\_Control](#_Music_Control_(0x04)) |  | V2.00 |  |
| 0x05 | [Change\_Device\_Name](#_Change_Device_Name_(0x05)) |  | V2.00 |  |
| 0x06 | [Change\_PIN\_Code](#_Change_PIN_Code_(0x06)_1) |  | V2.00 |  |
| 0x07 | [BTM\_Parameter\_Setting](#_BTM_Parameter_Setting_(0x07)_1) |  | V2.00 |  |
| 0x08 | [Read\_BTM\_Version](#_Read_BTM_Version_(0x08)) | [Read\_BTM\_Version\_Reply](#_Read_BTM_Version_Reply_(0x18)_1) | V2.00 |  |
| 0x09 | Get\_PB\_By\_AT\_Cmd |  |  |  |
| 0x0A | [Vendor\_AT\_Command](#_Vendor_AT_Cmd_(0x0A)) | [Report\_Vendor\_AT\_Event](#_Report_Vendor_AT_Event_(0x1D)) | V2.00 |  |
| 0x0B | [AVC\_Vendor\_Dependent\_Cmd](#_AVC_Specific_Cmd_(0x0B)) | [AVC\_Vendor\_Dependent\_Response](#_AVC_Vendor_Dependent_Response_(0x1A) | V2.00 |  |
| 0x0C | [AVC\_Group\_Navigation](#_AVC_Group_Navigation_(0x0C)) |  | V2.00 |  |
| 0x0D | [Read\_Link\_Status](#_Read_Link_Status_(0x0D)) | [Read\_Link\_Status\_Reply](#_Read_Link_Status_Reply_(0x1E)) | V2.00 |  |
| 0x0E | [Read\_Paired\_Device\_Record](#_Read_Paired_Device_Record_(0x0E)) | [Read\_Paired\_Device\_Record\_Reply](#_Read_Paired_Device_Record_Reply_(0x) | V2.00 |  |
| 0x0F | [Read\_Local\_BD\_Address](#_Read_Local_BD_Address_(0x0F)) | [Read\_Local\_BD\_Address\_Reply](#_Read_Local_BD_Address_Reply_(0x20)) | V2.00 |  |
| 0x10 | [Read\_Local\_Device\_Name](#_Read_Local_Device_Name_(0x10)) | [Read\_Local\_Device\_Name\_Reply](#_Read_Local_Device_Name_Reply_(0x21)) | V2.00 |  |
| 0x11 | Set\_Access\_PB\_Method |  |  |  |
| 0x12 | [Send\_SPP/iAP\_Data](#_Send_SPP/iAP_Or_LE_Data_(0x12)) |  | V2.00 |  |
| 0x13 | [BTM\_Utility\_Function](#_BTM_Utility_Function_(0x13)) |  | V2.00 |  |
| 0x14 | [Event\_ACK](#_Event_Ack_(0x14)) |  | V2.00 |  |
| 0x15 | [Additional\_Profiles\_Link\_Setup](#_Additional_Profile_Link_Setup_(0x15) |  | V2.00 |  |
| 0x16 | [Read\_Linked\_Device\_Information](#_Read_Linked_Device_Information_(0x1) | [Read\_Linked\_Device\_Information\_Reply](#_Read_Linked_Device_Information_Repl_1) | V2.00 |  |
| 0x17 | [Profiles\_Link\_Back](#_Profile_Link_Back__(0x17)) |  | V2.00 |  |
| 0x18 | [Disconnect](#_Disconnect_(0x18)) |  | V2.00 |  |
| 0x19 | MCU\_Status\_Indication |  |  |  |
| 0x1A | [User\_Confirm\_SPP\_Req\_Reply](#_User_Confirm_SSP_Req_Reply_(0x1A)) |  | V2.00 |  |
| 0x1B | [Set\_HF\_Gain\_Level](#_Set_HF_Speaker_Gain_Level_(0x1B)) |  | V2.00 |  |
| 0x1C | [EQ\_Mode\_Setting](#_EQ_Mode_Setting_(0x1C)) | [EQ\_Mode\_Indication](#_EQ_Mode_Indication_(0x10)_1) | V2.00 |  |
| 0x1D | [DSP\_NR\_CTRL](#_DSP_NR_CTRL_(0x1D)) |  | V2.00 |  |
| 0x1E | [GPIO\_Control](#_GPIO_CTRL_(0x1E)) | [Report\_Input\_Signal\_Level](#_Report_Input_Signal_Level_(0x27)_1) | V2.00 |  |
| 0x1F | [MCU\_UART\_Rx\_Buffer\_Size](#_MCU_UART_Rx_Buffer_Size_(0x1F)) |  | V2.00 |  |
| 0x20 | [Voice\_Prompt\_Cmd](#_Voice_Prompt_Cmd_(0x20)) | [Report\_Voice\_Prompt\_Status](#_Report_Voice_Prompt_Status_(0x2A)_1) | V2.00 |  |
| 0x21 | MAP\_REQUEST |  |  |  |
| 0x22 | Security\_Bonding\_Req |  | V2.00 | V2.01 |
| 0x23 | [Set\_Overall\_Gain](#_Set_Overall_Gain_(0x23)) |  | V2.00 |  |
| 0x24 | [Read\_BTM\_Setting](#_Read_BTM_Setting_(0x24)) | [REPORT\_TYPE\_BTM\_SETTING](#_Report_Type_BTM_Settings_(0x2E)_1) | V2.00 |  |
| 0x25 | [Read\_BTM\_Batt\_CHG\_Status](#_Read_BTM_Battery_Charger_Status__(0) | [BTM\_Battery\_Status](#_BTM_Battery_Status_(0x0C)_1) [BTM\_Charging\_Status](#_BTM_Charging_Status_(0x0D)_1) | V2.00 |  |
| 0x26 | [MCU\_Update\_Cmd](#_MCU_Update_Cmd__(0x26)) |  | V2.00 |  |
| 0x27 | [REPORT\_BATTERY\_CAPACITY](#_Report_Battery_Capacity__(0x27)) |  | V2.00 |  |
| 0x28 | [LE\_ANCS\_Service\_Cmd](#_LE_ANCS_Service_Cmd_(0x28)) |  | V2.00 |  |
| 0x29 | [LE\_Signaling\_Cmd](#_READ_FEATURE_LIST_(0x3A)) |  | V2.00 |  |
| 0x2A | [nSPK Vendor Cmd](#_nSPK_Vendor_Cmd_(0x2A)) |  | V2.00 |  |
| 0x2B | [Read\_NSPK\_Link\_Status](#_Read_nSPK_Link_Status__(0x2B)) | [Report\_nSPK\_Link\_Status](#_Report_nSPK_Link_Status_(0x33)_1) | V2.00 |  |
| 0x2C | [NSPK\_Sync\_Audio\_Effect](#_nSPK_Sync_Audio_Effect_(0x2C)) |  | V2.00 |  |
| 0x2D | [LE\_GATT\_CMD](#_LE_GATT_CMD_(0x2D)) |  | V2.00 |  |
| 0x2F | [LE\_App\_CMD](#_LE_App_Cmd_(0x2F)) |  | V2.00 |  |
| 0x30 | [DSP\_RUNTIME\_PROGRAM](#_DSP_Runtime_Program__(0x30)_1) |  | V2.00 |  |
| 0x31 | [Read\_Vendor\_EEPROM\_Data](#_Read_Vendor_Eeprom_Data__(0x31)) | [Report\_Vendor\_EEPROM\_Data](#_Report_Vendor_EEPROM_Data_(0x37)_1) | V2.00 |  |
| 0x32 | [Query IC version information](#_Read_IC_Version_Info_(0x32)) | [Report\_IC\_Ver\_Info](#_Report_IC_Ver_Info_(0x38)_1) | V2.00 |  |
| 0x33 | [Voice\_Prompt\_Ind\_Cmd](#_Voice_Prompt_Ind_Cmd_(0x33)) |  | V2.00 |  |
| 0x34 | [Read\_BTM\_Link\_Mode](#_Read_BTM_Link_Mode_(0x34)) | [Report\_BTM\_Link\_Mode](#_Report_BTM_Link_Mode_(0x3A)_1) | V2.00 |  |
| 0x35 | [Configure\_Vendor\_Parameter](#_Configure_Vendor_Parameter_(0x35)) |  | V2.00 |  |
| 0x36 | [DSP\_Dedicated\_Cmd](#_Configure_Vendor_Parameter_(0x36)) | [DSP\_Dedicated\_Event](#_DSP_Dedicated_Event_(0x3B)) | V2.00 |  |
| 0x37 | [nSPK Exchange\_Link\_Info\_Cmd](#_nSPK_Exchange_Link_Info_Cmd_(0x37)) |  | V2.00 |  |
| 0x38 | UART\_CMD\_NSPK\_SET\_GIAC |  | V2.00 |  |
| 0x39 | READ\_FEATURE\_LIST | [Report\_Read\_Feature\_List\_Reply](#_Report_Read_Feature_List_Reply_(0x4) | V2.00 |  |
| 0x3A | [Personal\_MSPK\_GROUP\_Control](#_GROUP_MASTER_ADDR_(0x3A)) |  | V2.00 |  |
| 0x3B | [UART\_CMD\_TEST\_DEVICE](#_UART_CMD_TEST_DEVICE_(0x3B)) |  | V2.00 |  |
| 0x40 | Reserved for internal test |  | V2.01 |  |

#### UART Event

|  |  |  |  |
| --- | --- | --- | --- |
| UART Event | | Support version | |
| OP code | Event | Start | End |
| [0x00](#_Command_Ack_(0x00)) | Command\_ACK | V2.00 |  |
| [0x01](#_BTM_Status_(0x01)) | BTM\_Status | V2.00 |  |
| [0x02](#_Call_Status_(0x02)) | Call\_Status | V2.00 |  |
| [0x03](#_Caller_ID_(0x03)) | Caller\_ID | V2.00 |  |
| [0x04](#_SMS_Received_Indication_(0x04)) | SMS\_Received\_Indication | V2.00 |  |
| [0x05](#_Missed_Call_Indication_(0x05)) | Missed\_Call\_Indication | V2.00 |  |
| [0x06](#_Phone_Max_Battery_Level_(0x06)) | Phone\_Max\_Battery\_Level | V2.00 |  |
| [0x07](#_Phone_Current_Battery_Level_(0x07)) | Phone\_Current\_Battery\_Level | V2.00 |  |
| [0x08](#_Roaming_Status_(0x08)) | Roaming\_Status | V2.00 |  |
| [0x09](#_Phone_Max_Signal_Strength_Level_(0x) | Phone\_Max\_Signal\_Strength\_Level | V2.00 |  |
| [0x0A](#_Phone_Current_Signal_Strength_Level) | Phone\_Current\_Signal\_Strength\_Level | V2.00 |  |
| [0x0B](#_Phone_Service_Status_(0x0B)) | Phone\_Service\_Status | V2.00 |  |
| [0x0C](#_BTM_Battery_Status_(0x0C)) | BTM\_Battery\_Status | V2.00 |  |
| [0x0D](#_BTM_Charging_Status_(0x0D)) | BTM\_Charging\_Status | V2.00 |  |
| [0x0E](#_Reset_To_Default_(0x0E)) | Reset\_To\_Default | V2.00 |  |
| [0x0F](#_Report_HF_Gain_Level_(0x0F)) | Report\_HF\_Gain\_Level | V2.00 |  |
| [0x10](#_EQ_Mode_Indication_(0x10)) | EQ\_Mode\_Indication | V2.00 |  |
| 0x11 | PBAP\_Missed\_Call\_History |  |  |
| 0x12 | PBAP\_Received\_Call\_History |  |  |
| 0x13 | PBAP\_Dialed\_Call\_History |  |  |
| 0x14 | PBAP\_Combine\_Call\_History |  |  |
| 0x15 | Phonebook\_Contacts |  |  |
| 0x16 | PBAP\_Access\_Finish |  |  |
| [0x17](#_Read_Linked_Device_Information_Repl) | Read\_Linked\_Device\_Information\_Reply | V2.00 |  |
| [0x18](#_Read_BTM_Version_Reply_(0x18)) | Read\_BTM\_Version\_Reply | V2.00 |  |
| [0x19](#_Call_List_Report_(0x19)) | Call\_List\_Report | V2.00 |  |
| [0x1A](#_AVC_Vendor_Dependent_Response_(0x1A) | AVC\_Specific\_Rsp | V2.00 |  |
| [0x1B](#_BTM_Utility_Req_(0x1B)) | BTM\_Utility\_Req | V2.00 |  |
| [0x1C](#_Vendor_AT_Cmd_Rsp_(0x1C)) | Vendor\_AT\_Cmd\_Reply | V2.00 |  |
| [0x1D](#_Report_Vendor_AT_Event_(0x1D)) | Report\_Vendor\_AT\_Event | V2.00 |  |
| [0x1E](#_Read_Link_Status_Reply_(0x1E)) | Read\_Link\_Status\_Reply | V2.00 |  |
| [0x1F](#_Read_Paired_Device_Record_Reply_(0x) | Read\_Paired\_Device\_Record\_Reply | V2.00 |  |
| [0x20](#_Read_Local_BD_Address_Reply_(0x20)) | Read\_Local\_BD\_Address\_Reply | V2.00 |  |
| [0x21](#_Read_Local_Device_Name_Reply_(0x21)) | Read\_Local\_Device\_Name\_Reply | V2.00 |  |
| [0x22](#_Reprt_SPP/iAP/LE_Data_(0x22)) | Report\_SPP/iAP\_Data | V2.00 |  |
| [0x23](#_Reprt_Link_Back_Status_(0x23)) | Report\_Link\_Back\_Status | V2.00 |  |
| [0x24](#_Ringtone_Status_Indication_(0x24)) | REPORT\_RING\_TONE\_STATUS | V2.00 |  |
| [0x25](#_User_Confirm_SSP_Req_(0x25)) | User\_Confrim\_SSP\_Req | V2.00 |  |
| [0x26](#_Report_AVRCP_Volume_Ctrl_(0x26)) | Report\_AVRCP\_Vol\_Ctrl | V2.00 |  |
| [0x27](#_Report_Input_Signal_Level_(0x27)) | Report\_Input\_Signal\_Level | V2.00 |  |
| [0x28](#_Report_iAP_Info_(0x28)) | Report\_iAP\_Info | V2.00 |  |
| [0x29](#_Report_AVRCP_ABS_Volume_Level_(0x29) | REPORT\_AVRCP\_ABS\_VOL\_CTRL | V2.00 |  |
| [0x2A](#_Report_Voice_Prompt_Status_(0x2A)) | Report\_Voice\_Prompt\_Status | V2.00 |  |
| 0x2B | Report\_MAP\_Data |  |  |
| 0x2C | Security\_Bonding\_Res | V2.00 |  |
| [0x2D](#_Report_Type_Codec_(0x2D)) | Report\_Type\_Codec | V2.00 |  |
| [0x2E](#_Report_Type_BTM_Settings_(0x2E)) | Report\_Type\_BTM\_Setting | V2.00 |  |
| [0x2F](#_Report_MCU_Update_Reply_(0x2F)) | Report\_MCU\_Update\_Reply | V2.00 |  |
| [0x30](#_Report_BTM_Initial_Status_(0x30)) | Report\_BTM\_Initial\_Status | V2.00 |  |
| [0x31](#_LE_ANCS_Service_Event_(0x31)) | LE\_ANCS\_Service\_Event | V2.00 |  |
| [0x32](#_LE_Signaling_Event_(0x32)) | LE\_Signaling\_Event | V2.00 |  |
| [0x33](#_Report_nSPK_Link_Status_(0x33)) | Report\_nSPK\_Link\_Status | V2.00 |  |
| [0x34](#_Report_nSPK_Vendor_Event_(0x34)) | Report\_nSPK\_Vendor\_Event | V2.00 |  |
| [0x35](#_Report_nSPK_Audio_Setting_(0x35)) | Report\_nSPK\_Audio\_Setting | V2.00 |  |
| [0x36](#_Report_Sound_Effect_Status_(0x36)) | Report\_Sound\_Effect\_Status | V2.00 |  |
| [0x37](#_Report_Vendor_EEPROM_Data_(0x37)) | Report\_Vendor\_EEPROM\_Data | V2.00 |  |
| [0x38](#_Report_IC_Ver_Info_(0x38)) | REPORT\_IC\_VERSION\_INFO | V2.00 |  |
| [0x39](#_Report_Customer_GATT_Attribute_Data) | REPORT\_LE\_GATT\_EVENT | V2.00 |  |
| [0x3A](#_Report_BTM_Link_Mode_(0x3A)) | Report\_BTM\_Link\_Mode | V2.00 |  |
| [0x3B](#_DSP_Dedicated_Event_(0x3B)) | DSP\_Dedicated\_Event | V2.00 |  |
| [0x3C](#_Report_nSPK_MISC_Event_(0x3C)) | Report\_nSPK\_MISC\_Event | V2.00 |  |
| [0x3D](#_Report_nSPK_Exchange_Link_info_(0x3) | Report\_nSPK\_Exchange\_Link\_Info | V2.00 |  |
| [0x3E](#_Report_Customized_Information_(0x3E) | Report Customized\_Information | V2.00 |  |
| [0x3F](#_Report_Read_Feature_List_Reply_(0x4) | Report\_CSB\_CLK | V2.00 |  |
| [0x40](#_Report_Read_Feature_List_Reply_(0x4) | Report\_Read\_Feature\_List\_Reply | V2.00 |  |
| [0x41](#_REPORT_TEST_RESULT_REPLY_(0x41)) | REPORT\_TEST\_RESULT\_REPLY | V2.00 |  |
| 0x50 | Reserved for internal use | V2.01 |  |

## UART Commands

### Make\_Call (0x00)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Make\_Call | 0x00 | Data\_Base\_Index,  Phone\_Number | [Call\_Status](#_Call_Status_(0x02)_1) |

**Description:**

This command is used to trigger HF action for making an outgoing call. Send out a standard AT command intended for placing a call to a phone number. Only voice calls are covered in this specification. Refer to Section 6.2 in [[1]](#_References).

DSPK

MCU

ATD+Phone\_Number

Call\_status (0x02)

**Precondition:**

HF should be in connected state.

**Command Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for dedicate link |
| 0x01 | database 1 for dedicate link |

Phone\_Number: Length: 19 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX | ASCII code of the phone number. The max length of phone number is 19 |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection exist |
| 0x03 | Parameter error | No phone number |

[[Return to Command Table]](#_UART_Command)

### Make\_Extension\_Call (0x01)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Make\_Extension\_Call | 0x01 | Data\_Base\_Index, Extension\_Number |  |

**Description:**

This command is used to trigger HF action for making an extension call number. During an ongoing call, the HF transmits the AT+VTS command to instruct the AG to transmit a specific DTMF code to its network connection.

**Precondition:**

1. An ongoing Service Level Connection between the AG and the HF shall exist.
2. An ongoing call in the AG exists.

MCU

DSPK

Outgoing call SLC connection

AT+VTS=….

OK

**Command Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for dedicate link |
| 0x01 | database 1 for dedicate link |

Extension\_Number: Length: 10 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX | ASCII code of the Extension phone number. The max length of phone number is 10 bytes |

**Return Parameters:**

Status: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Command succeeded |
| 0x01 – 0xFF | Command failed. See listing of Error Codes. |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection exist |

[[Return to Command Table]](#_UART_Command)

### Event\_Filter\_Setting (0x03)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Event\_Filter\_Setting | 0x03 | Event\_Filter |  |

**Description:**

This command is used to control which events are to be filtered for the Host MCU. If the bit in the Event\_Filter\_Setting is set to one, then the event associated with that bit will not be reported. The Host MCU has to deal with each event that is generated by BTM. The event filter setting allows the Host MCU to control which events will interrupt it. The default values of Event\_Filter Bit are all zero.

**Precondition:**

None.

**Command Parameters:**

Event\_Filter: Length: 4 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXXXXXX | Refer the Event\_Filter\_Table  Bit is 0:BTM will report this event  Bit is 1:BTM will not report this event |

Event\_Filter\_Table: Length: 4 Bytes

|  |  |
| --- | --- |
| Byte 0 | Parameter Description |
| Bit 0 | Reserved |
| Bit 1 | Reserved |
| Bit 2 | Call Status |
| Bit 3 | Incoming Call Number or Caller ID |
| Bit 4 | SMS Received |
| Bit 5 | Missed Call |
| Bit 6 | Max Cell Phone Battery Level |
| Bit 7 | Current Cell Phone Battery Level |

|  |  |
| --- | --- |
| Byte 1 | Parameter Description |
| Bit 0 | Cell Phone Roaming |
| Bit 1 | Max Cell Phone Signal Strength |
| Bit 2 | Current Cell Phone Signal Strength |
| Bit 3 | Cell Phone Service Status |
| Bit 4 | BTM Battery Level |
| Bit 5 | BTM Charging Status |
| Bit 6 | BTM Reset To Default Settings OK |
| Bit 7 | BTM DAC Gain Level |

|  |  |
| --- | --- |
| Byte 2 | Parameter Description |
| Bit 0 | EQ Mode |
| Bit 1 | Remote Device Friendly Name |
| Bit 2 | AVC Vendor Specific Response |
| Bit 3 | Unknown AT Command Result Code |
| Bit 4 | Page Status |
| Bit 5 | Ringtone Status |
| Bit 6 | Reserved |
| Bit 7 | Reserved |

|  |  |
| --- | --- |
| Byte 3 | Parameter Description |
| Bit 0 | Reserved |
| Bit 1 | Reserved |
| Bit 2 | Reserved |
| Bit 3 | Reserved |
| Bit 4 | Reserved |
| Bit 5 | Reserved |
| Bit 6 | Reserved |
| Bit 7 | Reserved |

[[Return to Command Table]](#_UART_Command)

### Music\_Control (0x04)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Music\_Control | 0x04 | Reserved, Action |  |

**Description:**

This command is used to trigger AVRCP commands for music control.

**Precondition:**

AVRCP should be active. If not, BTM will initiate the AVRCP connection if A2DP is active and no voice call in progress.

**Command Parameters:**

Reserved: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reserved |

Action: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Stop Fast Forward or Rewind |
| 0x01 | Fast Forward |
| 0x02 | Fast Forward With Repeat, Send Fast Forward Command For Every 812.5ms |
| 0x03 | Rewind |
| 0x04 | Rewind With Repeat, Send Rewind Command For Every 812.5ms |
| 0x05 | Play Command |
| 0x06 | Pause Command |
| 0x07 | Play Pause Toggle |
| 0x08 | Stop Command |
| 0x09 | Next song |
| 0x0A | Previous song |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No AVRCP connection exist |
| 0x03 | Parameter error | Incorrect Action value |

[[Return to Command Table]](#_UART_Command)

### Change\_Device\_Name (0x05)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Change\_Device\_Name | 0x05 | BT\_Device\_Name |  |

**Description:**

This command is used to change the device name of BTM.

Note: This command does not update the device name in the E2PROM.So new name set by this command will not be effective after the power cycle.

**Precondition:**

None.

**Command Parameters:**

BT\_Device\_Name: Length: 32 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX | Bluetooth device name |

[[Return to Command Table]](#_UART_Command)

### Change\_PIN\_Code (0x06)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Change\_PIN\_Code | 0x06 | PIN Code |  |

**Description:**

This command is used to change the BT PIN code of BTM used. Host MCU shall assert this command before BTM into pairing mode.

**Precondition:**

None.

**Command Parameters:**

PIN Code: Length: 4 Octets

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX | 4 digits number by ASCII format. |

[[Return to Command Table]](#_UART_Command)

### BTM\_Parameter\_Setting (0x07)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| BTM\_Parameter\_Setting | 0x07 | Parameter, Value1, Value2… |  |

**Description:**

This command is used to set the specific parameters that are listed in the below table of BTM.

**Precondition:**

None

**Command Parameters:**

Parameter: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | To Set Pairing Timeout Value |
| 0x01 | To Set Supported Codec Type(This change will update the e2prom) |
| 0x02 | To Enable/Disable BTM Standby Mode (This change will update the e2prom) |
| 0x03 | To Set The Recharging Battery Capacity Threshold |
| 0x04 | To Set Supported Profile |
| 0x05 | Set SBC bitpool setting : this should be set before A2DP connection established |
| 0x06 | Setting iAP2 serial number (This change will update the e2prom) |
| Others | Reserved |

Default the settings won’t update to EEPROM.

Value1 for Parameter 0x00: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Pairing Timeout Disabled |
| 0xXX | Pairing Timeout Value in the unit of 30.08secs. |

Value1 for Parameter 0x01: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| Bit 1  Bit 2 | Bit Mask:  should be set to1 for Enabling AAC  should be set to 1 for Enabling Vendor Specific Codec  By default SBC codec is enabled |

Value1 for Parameter 0x02: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | To Disable BTM Standby Mode |
| 0x01 | To Enable BTM Standby Mode |

Value1 for Parameter 0x03: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Range in 0~100 and Unit In Percentage |

Value1 for Parameter 0x04: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description (Bit Mask: Set to 1 to enable) |
| Bit 0 | HSP |
| Bit 1 | HFP |
| Bit 2 | A2DP |
| Bit 3 | AVRCP CT |
| Bit 4 | AVRCP TG |
| Bit 5 | SPP |
| Bit 6 | iAP |
| Bit 7 | Reserved |

Value1 for Parameter 0x05: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Maximal bitpool setting  Range : 0~250 |

Value2 for Parameter 0x05: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Minimal bitpool setting  Range : 0~250 |

Value1 for Parameter 0x06: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | iAP2 serial number Length  Max Length: 16 |

Value2-N for Parameter 0x06: Length: (N-2) Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| xxxx | iAP2 serial number (ASCII string) |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x03 | Parameter error | Incorrect Parameter value  Parameter 0x03: check battery setting range  Parameter 0x05: check bit-pool range |

[[Return to Command Table]](#_UART_Command)

### Read\_BTM\_Version (0x08)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Read\_BTM\_Version | 0x08 | Type | [Read\_BTM\_Version\_Reply](#_Read_BTM_Version_Reply_(0x18)_1) |

**Description:**

This command is used to query the supported UART command set version or FW version of BTM.

**Precondition:**

None.

**Command Parameters:**

Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | To Query Supported UART Command Set Version |
| 0x01 | To Query BTM FW Version |

[[Return to Command Table]](#_UART_Command)

### Vendor\_AT\_Cmd (0x0A)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Vendor\_AT\_Cmd | 0x0A | Data\_Base\_Index,Cmd\_Payload | [Report\_Vendor\_AT\_Event](#_Report_Vendor_AT_Event_(0x1D)) |

**Description:**

This command is used to send any vendor specific AT command.

**Precondition:**

HF should be in connected state and there should not be any vendor specific AT command already in progress.

**Command Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for dedicate link |
| 0x01 | database 1 for dedicate link |

Cmd\_Payload: Length: N Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX.. | The AT command ASCII string excluding the "AT".  For example : If MCU want to send "AT+ABCDE", the command payload should be "+ABCDE" |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | 1. No HF connection exist 2. Vendor specific AT command in progress |

[[Return to Command Table]](#_UART_Command)

### AVC\_Vendor\_Dependent\_Cmd (0x0B)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| AVC\_Vendor\_Dependent\_Cmd | 0x0B | Data\_Base\_Index, Avc\_Cmd\_Payload | [AVC\_Vendor\_Dependent\_Response](#_AVC_Vendor_Dependent_Response_(0x1A) |

**Description:**

This command is used to send vendor dependent AVC type commands. Only single packet type is supported for this command. Refer the Avc\_Cmd\_Payload table given below for the supported PDU IDs. Detail setting needs to refer to AVRCP Specification [[4]](#_References)

**Precondition:**

AVRCP should be active.

**Command Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for dedicate link |
| 0x01 | database 1 for dedicate link |

Avc\_Cmd\_Payload: Length: N Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| Byte 0  0x10  0x11  0x12  0x13  0x14  0x15  0x16  0x17  0x18  0x20  0x30  0x31  0x40  0x41 | AVRCP1.3 AVC Vendor Dependent Command PDU ID  Get Capabilities  List Player Application Setting Attributes  List Player Application Setting Values  Get Current Player Application Setting Value  Set Player Application Setting Value  Get Player Application Setting Attribute Text  Get Player Application Setting Value Text  Inform Displayable Character Set  Inform Battery Status Of CT  Get Element Attributes  Get Play Status  Register Notification  Request Continuing Response  Abort Continuing Response |
| Byte 1 | 0x00 |
| Byte 2-3 | Parameter Length |
| Byte 4-N | Parameter |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No AVRCP connection |

[[Return to Command Table]](#_UART_Command)

### AVC\_Group\_Navigation (0x0C)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| AVC\_Group\_Navigation | 0x0C | Data\_Base\_Index, Navigation\_Type |  |

**Description:**

The basic group navigation commands have a similar behavior as the Forward and Backward commands, but instead of navigating to the next/previous song they are used to navigate to the first song in the next/previous group. Detail setting needs to refer to AVRCP Specification [[4]](#_References)

**Precondition:**

AVRCP should be active.

**Command Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for dedicate link |
| 0x01 | database 1 for dedicate link |

Navigation\_Type: Length: N Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Next Group  This function is used to move to the first song in the next group. |
| 0x01 | Previous Group  This function is used to move to the first song in the previous group. |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No AVRCP connection |

[[Return to Command Table]](#_UART_Command)

### Read\_Link\_Status (0x0D)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Read\_Link\_Status | 0x0D | Reserved | [Read\_Link\_Status\_Reply](#_Read_Link_Status_Reply_(0x1E)) |

**Description:**

This command is used to query the device state, profile link status, playback status and streaming status. Refer the Read\_Link\_Status\_Reply event for the detailed information.

**Precondition:**

None.

**Command Parameters:**

Reserved: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reserved |

[[Return to Command Table]](#_UART_Command)

### Read\_Paired\_Device\_Record (0x0E)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Read\_Paired\_Device\_Record | 0x0E | Reserved | [Read\_Paired\_Device\_Record\_Reply](#_Read_Paired_Device_Record_Reply_(0x) |

**Description:**

This command is used to read the paired device information from BTM. The information will have link priority (1 is the highest (newest device) and 4 is the lowest (oldest device) ) and BD address of the paired devices.

**Precondition:**

None.

**Command Parameters:**

Reserved: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reserved |

[[Return to Command Table]](#_UART_Command)

### Read\_Local\_BD\_Address (0x0F)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Read\_Local\_BD\_Address | 0x0F | Reserved | [Read\_Local\_BD\_Address\_Reply](#_Read_Local_BD_Address_Reply_(0x20)) |

**Description:**

This command is used to read the local BD address.

**Precondition:**

None.

**Command Parameters:**

Reserved: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reserved |

[[Return to Command Table]](#_UART_Command)

### Read\_Local\_Device\_Name (0x10)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Read\_Local\_Device\_Name | 0x10 | Reserved | [Read\_Local\_Device\_Name\_Reply](#_Read_Local_Device_Name_Reply_(0x21)) |

**Description:**

This command is used to read the local device name.

**Precondition:**

None.

**Command Parameters:**

Reserved: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reserved |

[[Return to Command Table]](#_UART_Command)

### Send\_SPP/iAP\_Or\_LE\_Data (0x12)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Send\_SPP/iAP\_Or\_LE\_Data | 0x12 | Channel\_Index,Type,Total\_Length,Payload\_Length,Payload |  |

**Description:**

This command is used to send the SPP/iAP or LE data to remote BT devices.

**Precondition:**

SPP should be in connected state or

LE transparent service should be active.

**Command Parameters:**

Channel\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | MCU receives channel index information while SPP connected is established. If the connection is iAP or iAP2, MCU will receives the correct channel index when Report\_iAP\_Info is received because iAP session is created until that time.  bit0~1: bluetooth connection index(data base index). Range from 0~3.  bit 2: LE connection indicator  bit3~5: rfcomm index. Range from 0 ~ 7.  bit6~7: iAP session index. 1 is 1st session and 2 is 2nc session. 0 means not a iAP / iAP2 connection |

Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Single Packet |
| 0x01 | Fragmented Start Packet |
| 0x02 | Fragmented Continue Packet |
| 0x03 | Fragmented End Packet |

Total\_Length: Length: 2 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX | Total Payload Length |

Payload\_Length: Length: 2 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX | Payload Length In This Packet |

Payload: Length: N Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX | The Payload In This Packet |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | 1. No SPP connection 2. BLE Transparent Service disable |
| 0x03 | Parameter error | Incorrect parameter |
| 0x04 | BTM busy | BTM is busy |
| 0x05 | BTM Memory Full | TX buffer is full |

[[Return to Command Table]](#_UART_Command)

### BTM\_Utility\_Function (0x13)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| BTM\_Utility\_Function | 0x13 | Utility\_Function\_Type,Parameter |  |

**Description:**

This command is used to indicate BTM to execute the specific utility function. Supported functions list in below table.

**Precondition:**

None**.**

**Command Parameters:**

Utility\_Function\_Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Host MCU ask BTM to process NFC detected function. |
| 0x01 | To Enable/Disable in-built Aux Line In Function  If in-built Aux Line In Function is enabled then once line in is detected , BTM will process built-in aux line-in detection procedure |
| 0x02 | To generate the specific tone |
| 0x03 | To make BTM inactive or active |
| 0x04 | To indicate charger adaptor status |
| 0x05★ | To indicate BTM that remote device supports TTS engine. The BTM shall disable internal TTS engine. |
| 0x06 | To update partial EEPROM data which are related to part of audio configuration. |
| 0x07 | Voice prompt for the given version number. |
| 0x08 | For NSPK, MCU notifies the BTM current power condition |
| 0x09 | To update vendor EEPROM data |
| 0x0A | For NSPK, To inform Master that certain status has been changed in Slave side |
| 0x0B | To Read Serial number .For this command, MCU will receive event Report\_Vendor\_EEPROM\_Data with report data 16 bytes. |
| 0x0C★ | To switch audio channel |
| 0x0D★ | Customized MCU report :  MCU Report specified information the following parameter |
| 0x0E★ | Customized MCU request:  MCU request specified information by the following parameter.  BTM replies the specified information by E3E |

★ Customize commands default are disable. Please check the firmware release note to identify which customize commands support.

Parameter for Utility\_Function\_Type 0x00: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reserved |

**Function description:**

This function is used for MCU to ask BTM to process “NFC detected” function.

If BTM is under OFF state, BTM will power on.

If BTM is under PAIRING state, BTM will enter STANDBY state.

If BTM is under other state, BTM will enter PAGE SCAN ENABLE state.

Parameter for Utility\_Function\_Type 0x01: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Line in is not controlled by MCU |
| 0x01 | Line in is controlled by MCU |

Parameter for Utility\_Function\_Type 0x02: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Tone Type |

Tone Type: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Parameter Description | Sub Parameter |
| 0x00 | N/A | 0 |
| 0x01 | 200Hz | 100msec |
| 0x02 | 500Hz | 100msec |
| 0x03 | 1KHz | 100msec |
| 0x04 | 1.5KHz | 100msec |
| 0x05 | 2KHz | 100msec |
| 0x06 | 200Hz | 500msec |
| 0x07 | 500Hz | 500msec |
| 0x08 | 1KHz | 500msec |
| 0x09 | 1.5KHz | 500msec |
| 0x0A | 2KHz | 500msec |
| 0x0B | 200Hz / mute / 200Hz | 100msec for each tone |
| 0x0C | 500Hz / mute /500Hz | 100msec for each tone |
| 0x0D | 1KHz / mute /1KHz | 100msec for each tone |
| 0x0E | 1.5KHz / mute /1.5KHz | 100msec for each tone |
| 0x0F | 2KHz / mute / 2KHz | 100msec for each tone |
| 0x10 | 200Hz / mute /200Hz / mute /200Hz | 100msec for each tone |
| 0x11 | 500Hz / mute /500Hz / mute /500Hz | 100msec for each tone |
| 0x12 | 1KHz / mute /1KHz / mute /1KHz | 100msec for each tone |
| 0x13 | 1.5KHz / mute /1.5KHz / mute /1.5KHz | 100msec for each tone |
| 0x14 | 2KHz / mute /2KHz / mute /2KHz | 100msec for each tone |
| 0x15 | 200Hz / mute /200Hz / mute /200Hz mute / / 200Hz | 100msec for each tone |
| 0x16 | 500Hz / mute /500Hz / mute /500Hz / mute /500Hz | 100msec for each tone |
| 0x17 | 1KHz / mute /1KHz / mute /1KHz / mute /1KHz | 100msec for each tone |
| 0x18 | 1.5KHz / mute /1.5KHz / mute /1.5KHz / mute /1.5KHz | 100msec for each tone |
| 0x19 | 2KHz / mute /2KHz / mute /2KHz / mute /2KHz | 100msec for each tone |
| 0x1A | 500Hz / 400Hz / 300Hz / 200Hz | 50msec for each tone |
| 0x1B | 200Hz / 300Hz / 400Hz / 500Hz | 50msec for each tone |
| 0x1C | 400Hz / 300Hz | 150msec for each tone |
| 0x1D | 300Hz / 400Hz | 150msec for each tone |
| 0x1E | 300Hz / mute / 400Hz / mute / 500Hz / mute / 1000Hz | 100msec for each tone |
| 0x1F | 1000Hz / mute /500Hz / mute / 400Hz / mute /300Hz | 100msec for each tone |
| 0x20 | ROM build-in multi tone melody |  |
| 0x21 | ROM build-in multi tone melody |  |
| 0x22 | ROM build-in multi tone melody |  |
| 0x23 | ROM build-in multi tone melody |  |
| 0x24 | ROM build-in multi tone melody |  |
| 0x25 | ROM build-in multi tone melody |  |
| 0x26 | ROM build-in multi tone melody |  |

**Stored Voice prompt**

|  |  |
| --- | --- |
| Tone\_type | Voice prompt description |
| 0x80 | VP\_POWER\_ON |
| 0x81 | VP\_POWER\_OFF |
| 0x82 | VP\_PAIRING\_MODE |
| 0x83 | VP\_PAIRING\_COMPLETE |
| 0x84 | VP\_PAIRING\_NOT\_COMPLETE |
| 0x85 | VP\_CONNECTED |
| 0x86 | VP\_DISCONNECTED |
| 0x87 | VP\_INCOMING\_CALL |
| 0x88 | VP\_REJECT\_CALL |
| 0x89 | VP\_CALL\_END |
| 0x8A | VP\_VOICE\_DIAL |
| 0x8B | VP\_REDIAL |
| 0x8C | VP\_BATTERY\_L |
| 0x8D | VP\_BATTERY\_M |
| 0x8E | VP\_BATTERY\_H |
| 0x8F | VP\_CHARGING\_START |
| 0x90 | VP\_CHARGING\_OK |
| 0x91 | VP\_MAX\_VOL |
| 0x92 | VP\_MIN\_VOL |
| 0x93 | VP\_TONE\_SET |

Parameter for Utility\_Function\_Type 0x03: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | To force BTM into non-connectable mode |
| 0x01 | To resume BTM to normal mode |

Parameter for Utility\_Function\_Type 0x04: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Adaptor plugged in |
| Others | Adaptor unplugged |

Parameter for Utility\_Function\_Type 0x05: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x01 | To indicate BTM that remote device supported TTS engine. The BTM shall disable internal TTS engine. |
| Others | Reserved |

Parameter for Utility\_Function\_Type 0x06: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reserved |

Parameter for Utility\_Function\_Type 0x07:

|  |  |
| --- | --- |
| Value | Parameter Description |
| parameter[0] | Version length. Ex: v1.05 length is 3 |
| parameter[1~length] | Version value. Ex: v1.05 value is 1 0 5 |

Parameter for Utility\_Function\_Type 0x08: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Battery power with Low battery |
| 0x01 | Adaptor power with Low battery |
| 0x02 | Battery power |
| 0x03 | Adaptor power |

Parameter for Utility\_Function\_Type 0x09:

|  |  |
| --- | --- |
| Value | Parameter Description |
| parameter[0] | Vendor EEPROM offset |
| parameter[1] | Update length |
| parameter[2~length+1] | Update data |

Parameter for Utility\_Function\_Type 0x0A: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reserved |

Parameter for Utility\_Function\_Type 0x0B: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reserved |

Parameter for Utility\_Function\_Type 0x0C: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | L+R |
| 0x01 | L+L |
| 0x02 | R+R |
| 0x03 | (L+R)/2 |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x03 | Parameter error | Parameter incorrect |

[[Return to Command Table]](#_UART_Command)

### Event\_Ack (0x14)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Event\_Ack | 0x14 | Event\_ID |  |

**Description:**

This command is used for MCU to acknowledge the received BTM event.

**Precondition:**

None**.**

**Command Parameters:**

Event\_ID Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Event ID of the event which needs to be acknowledged |

[[Return to Command Table]](#_UART_Command)

### Additional\_Profile\_Link\_Setup (0x15)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Additional\_Profile\_Link\_Setup | 0x15 | Data\_Base\_Index,Linked\_Profile |  |

**Description:**

This command is used to initiate other profile connection based on the already existing link profiles. For example, you can initiate HF/HS profile connection if HF/HS is under standby mode and without connection and there is already has one A2DP/AVRCP/SPP connected profile.

**Precondition:**

ACL link should be connected and

A2DP or AVRCP or HF or SPP profile should be connected.

**Command Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database Index 0 that linked profile occupied |
| 0x01 | database Index 1 that linked profile occupied |

Linked\_Profile: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | To Initiate HF/HS Profile connection |
| 0x01 | To Initiate A2DP Profile connection |
| 0x02 | To Initiate iAp/SPP Profile connection |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No ACL link or  no any A2DP/AVRCP/HFP/HSP profile connected |

[[Return to Command Table]](#_UART_Command)

### Read\_Linked\_Device\_Information (0x16)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Read\_Linked\_Device\_Information | 0x16 | Data\_Base\_Index,  Type | [Read\_Linked\_Device\_Information\_Reply](#_Read_Linked_Device_Information_Repl_1) |

**Description:**

This command is used to retrieve the connected device profile information or local information based on the ‘Type’ parameter value**.**

**Precondition:**

For Type 0x00: ACL link should be connected

For Type 0x01: HF should be in connected state

For Type 0x02: SPP should be in connected state

For Type 0x03: AVRCP should be in connected state

For Type 0x04 and 0x05: None

**Command Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for dedicate link |
| 0x01 | database 1 for dedicate link |

Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Query Device Name |
| 0x01 | Query In-Band Ringtone Status |
| 0x02 | Query Whether remote device is iAP or standard SPP device |
| 0x03 | Query Whether remote device supports AVRCP 1.3 or not |
| 0x04 | Query HF/A2DP Gain |
| 0x05 | Query Line-In Gain |
| 0x06 | Reserved |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | Type 0x00: no ACL link  Type 0x01: no HF connection  Type 0x02: no SPP connection  Type 0x03: no AVRCP connection |
| 0x03 | Parameter error | Incorrect parameter |

[[Return to Command Table]](#_UART_Command)

### Profile\_Link\_Back (0x17)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Profile\_Link\_Back | 0x17 | Type, Device\_Index, Profile, BT\_Addr | [Report\_Link\_Back\_Status](#_Reprt_Link_Back_Status_(0x23)_1) |

**Description:**

This command is used to trigger the link for specific profiles to the devices in paired list.

**Precondition:**

Paired device list should exist.

**Command Parameters:**

Type: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Parameter Description | Sub Parameter |
| 0x00 | Connect to last device : if last device supports HF/HS, then initiate HF/HS connection, otherwise initiate A2DP connection | NA |
| 0x01 | Initiate HF/HS connection to last HF/HS device | NA |
| 0x02 | Initiate A2DP connection to last A2DP device | NA |
| 0x03 | Initiate SPP/iAP connection to last SPP/iAP device | NA |
| 0x04 | Initiate connection to dedicate device index with the profile specified by the Profile parameter | Device\_Index, Profile |
| 0x05 | Initiate connection to the specified BT Address | Device\_Index, Profile, BT\_Addr |
| 0x06 | initiate connection to multipoint mode | NA |

Device\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Device Index. Range is from 0 to 7 |

Profile: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | The profile determined by BTM's e2prom record. |
| Bit0 | HS profile. |
| Bit1 | HF profile. |
| Bit2 | A2DP profile |

BT\_Addr: Length:6 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXXXXXXXXXX | The Bluetooth address of the target device that BTM will trying to create a connection with |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | nSPK is creating connection  TYPE=0x01: The last device doesn’t support HF/HS/A2DP  Incorrect TYPE |

[[Return to Command Table]](#_UART_Command)

### Disconnect (0x18)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Disconnect | 0x18 | Disconnection\_Flag | [BTM\_Status](#_BTM_Status_(0x01)) |

**Description:**

This command is used to cancel the ongoing link back procedure or disconnect all the select linked profiles. BTM will disconnect ACL link if all of the profiles are disconnected.

**Precondition:**

None.

**Command Parameters:**

Disconnection\_Flag: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| Bit 0 | Cancel page before ACL connection has been created. |
| Bit 1 | Disconnect all of the HF connections |
| Bit 2 | Disconnect all of the A2DP connections |
| Bit 3 | Disconnect all of the SPP/BLE (if BLE enable) connections |
| Bit 4 | Disconnect all of the SPP connections |
| Bit 5 | Disconnect BLE connection |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x03 | Parameter error | Incorrect disconnection\_flag parameter |

[[Return to Command Table]](#_UART_Command)

### User\_Confirm\_SSP\_Req\_Reply (0x1A)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| User\_Confirm\_SSP\_Req\_Reply | 0x1A | Data\_Base\_Index, User\_Response |  |

**Description:**

This command is used to reply to a User\_Confirm\_SSP\_Req event and indicates that the user selected "yes" or "no".

**Precondition:**

ACL link should be connected.

**Command Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 that simple pairing is ongoing |
| 0x01 | database 1 that simple pairing is ongoing |

User\_Response: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | User Selected “yes” |
| 0x01 | User Selected “no” |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No ACL connection |

[[Return to Command Table]](#_UART_Command)

### Set\_HF\_Speaker\_Gain\_Level (0x1B)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Set\_HF\_Speaker\_Gain\_Level | 0x1B | Data\_Base\_Index, Gain\_Level |  |

**Description:**

This command is used to set HF Speaker gain of BTM.

**Precondition:**

HF should be in connected state.

**Command Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 that related to a dedicate HF device |
| 0x01 | database 1 that related to a dedicate HF device |

Gain\_Level: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 – 0x0F | HF Speaker Gain Level |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection |

[[Return to Command Table]](#_UART_Command)

### EQ\_Mode\_Setting (0x1C)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| EQ\_Mode\_Setting | 0x1C | EQ\_Mode, Reserved | [EQ\_Mode\_Indication](#_EQ_Mode_Indication_(0x10)_1) |

**Description:**

This command is used to set the EQ mode of BTM for audio playback.

**Precondition:**

EQ Mode should be enabled by using DSP configuration tool.

**Command Parameters:**

EQ\_Mode: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | EQ\_MODE\_OFF |
| 0x01 | EQ\_MODE\_SOFT |
| 0x02 | EQ\_MODE\_BASS |
| 0x03 | EQ\_MODE\_TREBLE |
| 0x04 | EQ\_MODE\_CLASSICAL |
| 0x05 | EQ\_MODE\_ROCK |
| 0x06 | EQ\_MODE\_JAZZ |
| 0x07 | EQ\_MODE\_POP |
| 0x08 | EQ\_MODE\_DANCE |
| 0x09 | EQ\_MODE\_RNB |
| 0x0A | EQ\_MODE\_USER1 |
| Others | Reserved |

Reserved: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reserved |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | DSP EQ mode not enable |

[[Return to Command Table]](#_UART_Command)

### DSP\_NR\_CTRL (0x1D)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| DSP\_NR\_CTRL | 0x1D | Cmd\_Type |  |

**Description:**

This command is used to set the noise reduction for voice link.

**Precondition:**

None.

**Command Parameters:**

Cmd\_Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x18 | ENABLE\_Mic\_NR |
| 0x19 | DISABLE\_Mic\_NR |
| 0x1B | ENABLE\_SPK\_NR |
| 0x1C | DISABLE\_SPK\_NR |
| Others | Reserved |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| V1.03 | | |
| **Value** | **Description** | **Condition** |
| 0x01 | Command disallow | DSP NR module disable |
| 0x03 | Parameter error | Parameter incorrect |

[[Return to Command Table]](#_UART_Command)

### GPIO\_CTRL (0x1E)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| GPIO\_CTRL | 0x1E | IO\_Ctrl\_Mask\_P0,  IO\_Ctrl\_Mask\_P1,  IO\_Ctrl\_Mask\_P2,  IO\_Ctrl\_Mask\_P3,  IO\_Setting\_P0,  IO\_Setting\_P1,  IO\_Setting\_P2,  IO\_Setting\_P3,  Output\_Value\_P0,  Output\_Value\_P1,  Output\_Value\_P2,  Output\_Value\_P3, | [Report\_Input\_Signal\_Level](#_Report_Input_Signal_Level_(0x27)_1) |

**Description:**

This command is used to control the specific GPIOs as input level detection or output level drive. For input level detection configuration, BTM will report the input signal level to MCU when input signal level is changed.

Note: For GPIOs configured as input, Report\_Input\_Signal\_Level event shall be send under two condition, first condition is when BTM receive input setting command, the second condition is when input GPIO input signal level is changed

**Precondition:**

None.

**Command Parameters:**

IO\_Ctrl\_Mask\_P0: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0bXXXXXXXX | Bit mask of P0 for IO control setting.  MCU set the bit value to 0 to control corresponding pin, otherwise set to 1. Bit 0: P0\_0 IO control setting mask. Bit 1: P0\_1 IO control setting mask. …… |

IO\_Ctrl\_Mask\_P1: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0bXXXXXXXX | Bit mask of P1 for IO control setting.  MCU set the bit value to 0 to control corresponding pin, otherwise set to 1. Bit 0: P1\_0 IO control setting mask. Bit 1: P1\_1 IO control setting mask. …… |

IO\_Ctrl\_Mask\_P2: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0bXXXXXXXX | Bit mask of P2 for IO control setting.  MCU set the bit value to 0 to control corresponding pin, otherwise set to 1. Bit 0: P2\_0 IO control setting mask. Bit 1: P2\_1 IO control setting mask. …… |

IO\_Ctrl\_Mask\_P3: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0bXXXXXXXX | Bit mask of P3 for IO control setting.  MCU set the bit value to 0 to control corresponding pin, otherwise set to 1. Bit 0: P3\_0 IO control setting mask. Bit 1: P3\_1 IO control setting mask. …… |

IO\_Setting\_P0: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0bXXXXXXXX | Setting P0 GPIO as input or output mode IO bit setting of P0 for input or output configuration. 0: input. 1: output  Bit 0: P0\_0 IO control setting. Bit 1: P0\_1 IO control setting. …… |

IO\_Setting\_P1: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0bXXXXXXXX | Setting P1 GPIO as input or output mode IO bit setting of P1 for input or output configuration. 0: input. 1: output  Bit 0: P1\_0 IO control setting. Bit 1: P1\_1 IO control setting. …… |

IO\_Setting\_P2: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0bXXXXXXXX | Setting P2 GPIO as input or output mode IO bit setting of P2 for input or output configuration. 0: input. 1: output  Bit 0: P2\_0 IO control setting. Bit 1: P2\_1 IO control setting. …… |

IO\_Setting\_P3: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0bXXXXXXXX | Setting P3 GPIO as input or output mode IO bit setting of P3 for input or output configuration. 0: input. 1: output  Bit 0: P3\_0 IO control setting. Bit 1: P3\_1 IO control setting. …… |

Output\_Value\_P0: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0bXXXXXXXX | Output level of P0.x GPIOs setting. It is used for IO\_Setting\_P0.x as output only. |

Output\_Value\_P1: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0bXXXXXXXX | Output level of P1.x GPIOs setting. It is used for IO\_Setting\_P1.x as output only. |

Output\_Value\_P2: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0bXXXXXXXX | Output level of P2.x GPIOs setting. It is used for IO\_Setting\_P2.x as output only. |

Output\_Value\_P3: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0bXXXXXXXX | Output level of P3.x GPIOs setting. It is used for IO\_Setting\_P3.x as output only. |

[[Return to Command Table]](#_UART_Command)

### MCU\_UART\_Rx\_Buffer\_Size (0x1F)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| MCU\_UART\_Rx\_Buffer\_Size | 0x1F | Max\_MCU\_UART\_Rx\_Buffer\_Size |  |

**Description:**

This command is used to indicate the BTM about the max UART Rx buffer size. The default value is 256 Bytes.

**Precondition:**

None.

**Command Parameters:**

Max\_MCU\_UART\_Rx\_Buffer\_Size: Length: 2 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX | The maximum UART receiver buffer size of Host MCU. |

[[Return to Command Table]](#_UART_Command)

### Voice\_Prompt\_Cmd (0x20)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Voice\_Prompt\_Cmd | 0x20 | Cmd\_Type, Parameter, Voice\_Data | [Report\_Voice\_Prompt\_Status](#_Report_Voice_Prompt_Status_(0x2A)_1) |

**Description:**

This command is used to control BTM voice prompt function. Host MCU shall set voice prompt control parameter first and then send voice prompt data after received the event Report\_Voice\_Prompt\_Status with ready status.

**Precondition:**

For Cmd\_Type 0x00: None

For Cmd \_Type 0x01: Voice\_Prompt\_Cmd(0x20) should be called with Cmd\_Type 0x00

For example:

1. 0x20 0x00 0x01
2. 0x20 0x01 0x00 “RAW data” which data size is smaller than 480 bytes

**Command Parameters:**

Cmd\_Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Voice Prompt Settings |
| 0x01 | Voice Prompt Data |
| Others | Reserved |

Parameter: for Cmd\_Type 0x00 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Stop to play voice prompt and skip the previous voice data. |
| 0x01 | initial, high priority: Force to stop current tone and clean tone queue then play the new tone |
| 0x02 | initial, low priority: just put the new tone into the tone queue |
| Others | reserved |

Parameter: for Cmd\_Type 0x01 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | single packet, if voice data size is less than 480 |
| 0x01 | fragmented start packet |
| 0x02 | fragmented continue packet |
| 0x03 | fragmented end packet |
| Others | reserved |

Voice\_Data: for Cmd\_Type 0x00 Length: N Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX | Invalid |

Voice\_Data: for Cmd\_Type 0x01 Length: N Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX | Voice Prompt Data |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | DSP Ringtone state is ready |
| 0x04 | BTM busy | BTM is busy |
| 0x03 | Parameter error | Parameter incorrect |

### Set\_Overall\_Gain (0x23)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Set\_Overall\_Gain | 0x23 | Data\_Base\_Index, Mask,Type,Gain1,Gain2,Gain3 |  |

**Description:**

This command is used to set overall gain that includes hf, a2dp and line\_in.

**Precondition:**

None.

**Command Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 |
| 0x01 | database 1 |

Mask: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Set '1' to indicate the gain need to change.  bit0:A2DP gain  bit1:HF gain  bit2:Line\_In gain  bit3:PCM gain |

Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x01 | volume up |
| 0x02 | volume down |
| 0x03 | set absolute gain level |
| 0x04 | set absolution gain value (only for AVRCP1.5) |
| 0x05 | sync volume setting from MCU |

Gain1: Effective when bit0 of mask is 1 or type is 4, 5 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | type=1,2  X  type=3 : A2DP gain level 0~15  0x00~0x0F  type=4,5 : A2DP absolute gain value 0%~100%  0x00~0x7F  Other types are reserved |

Gain 2: Effective when bit1 of mask is 1 or type is 5 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | type=1,2,4  X  type=3 : HF gain level 0~15  0x00~0x0F  type=5 : HF gain percentage 0%~100%  0x00~0x7F  Other types are reserved |

Gain 3: Effective when bit2 of mask is 1 or type is 5 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | type=1,2,4  X  type=3 : line in gain level 0~15  0x00~0x0F  type=5 : line in gain percentage 0%~100%  0x00~0x7F  Other types are reserved |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | Type 0x03: mask profiles aren’t connected.  Type 0x04: AVRCP is not connected. |
| 0x03 | Parameter error | Incorrect parameter:  Gain value is bigger than max gain setting. |

[[Return to Command Table]](#_UART_Command)

### Read\_BTM\_Setting (0x24)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Read\_BTM\_Setting | 0x24 | Setting\_Type, Reserved | [REPORT\_TYPE\_BTM\_SETTING](#_Report_Type_BTM_Settings_(0x2E)_1) |

**Description:**

This command is used to read setting status of BTM**.**

**Precondition:**

None.

**Command Parameters:**

Setting\_Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | read pairing timeout value |
| 0x01 | read supported codec type |
| 0x02 | read BTM Standby mode status |
| others | reserved |

Reserved: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reserved |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x03 | Parameter error | Incorrect parameter |

[[Return to Command Table]](#_UART_Command)

### Read\_BTM\_Battery\_Charger\_Status (0x25)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Read\_BTM\_Battery\_Charger\_Status | 0x25 | Type | [BTM\_Battery\_Status](#_BTM_Battery_Status_(0x0C)_1) [BTM\_Charging\_Status](#_BTM_Charging_Status_(0x0D)_1) |

**Description:**

This command is used to read either Battery or Charger status of BTM**.**

**Precondition:**

None.

**Command Parameters:**

Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | To Read Battery Status |
| 0x01 | To Read Charger Status |
| Others | Reserved |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x03 | Parameter error | Parameter incorrect. |

[[Return to Command Table]](#_Command_opcode_definition)

### MCU\_Update\_Cmd (0x26)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| MCU\_Update\_Cmd | 0x26 | Action |  |

**Description:**

This command is used to inform BTM about MCU boot loader status

**Precondition:**

None.

**Command Parameters:**

Action: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Update start and report the total size |
| 0x01 | MCU ready to receive data from BTM |
| 0x02 | MCU update finish |
| Others | Reserved |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x03 | Parameter error | Incorrect parameter |

[[Return to Command Table]](#_UART_Command)

### Report\_Battery\_Capacity (0x27)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Report\_Battery\_Capacity | 0x27 | Capacity |  |

**Description:**

This command is used to inform BTM about the system battery capacity (unit in percentage)

**Precondition:**

None.

**Command Parameters:**

Capacity: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Battery capacity, Range in 0~100 and Unit in percentage |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x03 | Parameter error | Parameter incorrect. |

[[Return to Command Table]](#_UART_Command)

### LE\_ANCS\_Service\_Cmd (0x28)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| LE\_ANCS\_Service\_Cmd | 0x28 | SubCommand\_Type, SubCommand\_Payload |  |

**Description:**

This command is used to access Apple Notification Center Service(ANCS), and get many kinds of notifications that are generated on iOS devices by Bluetooth low-energy link.

**Precondition:**

LE should be in connected state.

**Command Parameters:**

SubCommand\_Type : Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | ANCS Search |
| 0x01 | ANCS Subscribe |
| 0x02 | ANCS GetNotification Attribute |
| 0x03-0xFF | Reserved |

SubCommand\_Payload : for SubCommand\_Type 0x00 Length: 0 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| N/A | N/A |

SubCommand\_Payload : for SubCommand\_Type 0x01 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Un-subscribe ANCS |
| 0x01 | Subscribe ANCS |

SubCommand\_Payload : for SubCommand\_Type 0x02 Length: N Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
|  | [The format of a GetNotification Attribute command. [2]](#_References) |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No LE connection |
| 0x03 | Parameter error | For other than given SubCommand\_Type |
| 0x04 | BTM busy | BTM is busy |

[[Return to Command Table]](#_UART_Command)

### LE\_Signaling\_Cmd (0x29)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| LE\_Signaling\_Cmd | 0x29 | SubCommand\_Type, SubCommand\_Payload | [LE\_Signaling\_Event](#_LE_Signaling_Event_(0x32)_1) for SubCommand 0x00. |

**Description:**

This command is used to do the Bluetooth low energy signaling control.

**Precondition:**

None.

**Command Parameters:**

SubCommand\_Type : Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Query LE status |
| 0x01 | LE Advertising Control |
| 0x02 | LE Connection Parameters Update REQ |
| 0x03 | LE Advertising Interval Update |
| 0x04 | LE Advertising Type |
| 0x05 | LE Advertising Data |
| 0x06 | LE Scan Response Data |
| 0x07-0xFF | Reserved |

SubCommand\_Payload : for SubCommand\_Type 0x00 Length: 0 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| N/A | N/A |

SubCommand\_Payload : for SubCommand\_Type 0x01 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Disable Advertising |
| 0x01 | Enable Advertising |

SubCommand\_Payload : for SubCommand\_Type 0x02 Length: 8 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX(2 Bytes) | Connection Interval Minimum:  0x0006 to 0x0C80 (7.5ms ~ 4s), Minimum value for the connection event interval. |
| 0xXXXX(2 Bytes) | Connection Interval Maximum:  0x0006 to 0x0C80 (7.5ms ~ 4s), Maximum value for the connection event interval. |
| 0xXXXX(2 Byte) | Slave Latency:  Range: 0 to ((Supervision Timeout / (Connection Interval x2)) -1) and shall also be less than 500, The Slave Latency parameter defines the number of consecutive connection events that the slave device is not required to listen for the master. |
| 0xXXXX(2 Bytes) | Supervision :  Range: 0x000A to 0x0C80 (100ms ~ 32s), Supervision timeout for the LE Link |

SubCommand\_Payload : for SubCommand\_Type 0x03 Length: 2 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
|  | Advertising interval.  Range: 0x0020 ~ 0x4000.  Unit: 0.625ms |

SubCommand\_Payload : for SubCommand\_Type 0x04 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Connectable undirected advertising. |
| 0x01 | Reserved |
| 0x02 | Scannable undirected advertising |
| 0x03 | Non connectable undirected advertising. |

SubCommand\_Payload : for SubCommand\_Type 0x05 Length: 1 ~ 32 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
|  | Reserved. (1 byte)  Advertising data (max to 31 bytes). |

SubCommand\_Payload : for SubCommand\_Type 0x06 Length: 1 ~ 32 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
|  | Reserved. (1 byte)  Scan response data (max to 31 bytes). |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x03 | Parameter error | For other than given SubCommand\_Type |
| 0x01 | Command disallow | LE status incorrect |

[[Return to Command Table]](#_UART_Command)

### nSPK\_Vendor\_Cmd (0x2A)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| nSPK\_Vendor\_Cmd | 0x2A | SPK\_Index, Parameter |  |

**Description:**

This command is used to send the vendor nSPK command, after this command been sent to another SPKs, SPK may response OK, ERROR, or no response.



**SPK2**

**SPK1**

**Precondition:**

None.

**Command Parameters:**

SPK\_Index : Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | the slave SPK(SPK2/3) send vendor command to master SPK |
| 0x01 | master SPK(SPK1) send vendor command to the slave SPK(SPK2) |

Parameter: SPK\_Index :0x00 Length: N Bytes

|  |  |  |
| --- | --- | --- |
| SPK\_Index | Parameter Content | Parameter Description |
| 0x00 | Invalid payload data | they are used to show the status of slave SPK |
| 0x01~0x1F | Valid payload data. This data will be transfer to SPK Master and SPK Master’s MCU by E34 | |
| 0x20~0xFF | Invalid payload data. | |

Parameter: SPK\_Index :0x01

|  |  |  |
| --- | --- | --- |
| SPK\_Index | Parameter Content | Parameter Description |
| Byte[0] | High bytes of following data length | they are used to show the status of slave SPK |
| Byte[1] | Low bytes of following data length |  |
| Byte[2]~Byte[12] | Payload. 11 bytes are available |  |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x03 | Parameter error | SPK\_Inde 0x00: parameter incorrect. |

[[Return to Command Table]](#_UART_Command)

### Read\_nSPK\_Link\_Status (0x2B)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Read\_NSPK\_Link\_Status | 0x2B | Reserved | [Report\_nSPK\_Link\_Status](#_Report_nSPK_Link_Status_(0x33)_1) |

**Description:**

This command is used to query the nSPK link status.

**Precondition:**

None.

**Command Parameters:**

Reserved: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reserved |

[[Return to Command Table]](#_UART_Command)

### nSPK\_Sync\_Audio\_Effect (0x2C)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| NSPK\_Sync\_Audio\_Effect | 0x2C | audio\_effect\_type, audio\_effect\_value |  |

**Description:**

It is used for Master SPK to sync audio effect to Slave SPK in nSPK mode.

**Precondition:**

None.

**Command Parameters:**

audio\_effect\_type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | vendor defined audio effect type |

audio\_effect\_value: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | audio effect value |

[[Return to Command Table]](#_UART_Command)

### LE\_GATT\_CMD (0x2D)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| LE\_GATT\_CMD | 0x2D | Sub\_Cmd\_Type, Parameters |  |

**Description:**

Generic BLE GATT command for controlling MCU for specific GATT operations. This command is followed by sub commands for specific Gatt operations. The sub commands are detailed as below:

**Precondition:**

None.

#### Send\_Characteristic\_Value (0x00)

This command is used to send characteristic value to GATT client. It can be used for both notification and indication of char value to Gatt client.

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Sub-Op Code | Command Parameters | Return Event |
| Send\_Characteristic\_Value | 0x00 | Connection\_Handle, Characteristic\_Value\_Handle, Characteristic\_Value | Status |

**Command Parameters:**

Connection\_Handle: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Connection Handle |

Characteristic\_Value\_Handle: Length: 2 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Characteristic Value Handle |

Characteristic\_Value: Length: 1 to 20 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Characteristic Value |

#### Send\_Write\_Response (0x01)

This command is used to manually respond write request to the GATT client. DSPK will send “Write Response” with error code 0x00. Otherwise, “Error Response” with corresponding error code is sent. This command should only be sent after receiving write request.

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Sub-Op Code | Command Parameters | Return Event |
| Send\_Write\_Response | 0x01 | Connection\_Handle, Request\_Opcode, Attribute\_Handle, Error\_Code | Status |

**Command Parameters:**

Connection\_Handle: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Connection Handle |

Request\_Opcode: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Write Request. The op-code of the request to respond. |

Attribute\_Handle: Length: 2 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | The handle of the attribute which the request perform |

Error\_Code: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | No error. Send Write Response |
| 0x01 | Invalid Handle |
| 0x02 | Read Not Permitted |
| 0x03 | Write Not Permitted |
| 0x04 | Invalid PDU |
| 0x05 | Insufficient Authentication |
| 0x06 | Request Not Supported |
| 0x07 | Invalid Offset |
| 0x08 | Insufficient Authorization |
| 0x09 | Prepare Queue Full |
| 0x0A | Attribute Not Found |
| 0x0B | Attribute Not Long |
| 0x0C | Insufficient Encryption Key Size |
| 0x0D | Invalid Attribute Value Length |
| 0x0E | Unlikely Error |
| 0x0F | Insufficient Encryption |
| 0x10 | Unsupported Group Type |
| 0x11 | Insufficient Resources |
| 0x12 – 0x7F | Reserved |
| 0x80 – 0x9F | Application defined errors |
| 0xA0 – 0xDF | Reserved |
| 0xE0 – 0xFF | Common Profile and Service Error Codes |

#### Update\_Characteristic\_Value (0x02)

This command is used to update existing characteristic value of local attribute database

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Sub-Op Code | Command Parameters | Return Event |
| Update\_Characteristic\_Value | 0x02 | Characteristic\_Value\_Handle, Characteristic\_Value | Status |

**Command Parameters:**

Characteristic\_Value\_Handle: Length: 2 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | characteristic value handle |

Characteristic\_Value: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | characteristic value |

#### Read\_Local\_Characteristic\_Value (0x03)

This command is used to read existing characteristic value from local attribute database.

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Sub-Op Code | Command Parameters | Return Event |
| Read\_Local\_Characteristic\_Value | 0x03 | Characteristic\_Value\_Handle | Status  [Read\_Local\_Char\_Value\_Res](#_Read_Local_Char_Value_Res_(0x01)) |

**Command Parameters:**

Characteristic\_Value\_Handle: Length: 2 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | characteristic value handle |

#### Read\_Local\_All\_Primary\_Services (0x04)

This command is used to read all primary services from local attribute database.

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Sub-Op Code | Command Parameters | Return Event |
| Read\_Local\_All\_Primary\_Services | 0x04 | None | Status  [Discover\_All\_Primary\_Services\_Res](#_Discover_All_Primary_Services_Res_() |

**Command Parameters:**

None**.**

#### Read\_Local\_Specific\_Primary\_Service (0x05)

This command is used to read specific primary service from local attribute database.

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Sub-Op Code | Command Parameters | Return Event |
| Read\_Local\_Specific\_Primary\_Service | 0x05 | Service\_UUID | Status  Read\_Local\_Specific\_Primary\_Service\_Characteristic\_Res  Read\_Local\_All\_Char\_Descriptors\_Res |

**Command Parameters:**

Service\_UUID: Length: 2/16 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | 16-bit Bluetooth UUID or 128-bit UUID |

**Note: UUID should be in reverse byte order. Eg.: If UUID is 0x180D, then send as 0x0D18**

### LE\_App\_Cmd (0x2F)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| LE\_App\_Cmd | 0x2F | Sub\_Cmd\_Type, Parameter |  |

**Description:**

This command is used for LE application purpose.

**Precondition:**

For Sub\_Cmd\_Type 0x5D: LE state should be connected.

For Sub\_Cmd\_Type 0x5C: None

#### Set\_Device\_Name (0x5C)

This command is used to set the LE device name.

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Sub-Op Code | Command Parameters | Return Event |
| Set\_Device\_Name | 0x5C | Name\_Length,  Name\_String | Status |

**Command Parameters:**

Name\_Length: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Length of the LE device name |

Name\_String: Length: 1 to 20 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX.. | LE Device Name String |

#### Get\_Att\_MTU\_Size (0x5D)

This command is used to get the Att MTU size. Att MTU size will be notified to the host MCU by using the event REPORT\_LE\_GATT\_EVENT (0x39) with the sub event type Get\_Att\_MTU\_Size\_Res (0x05).

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Sub-Op Code | Command Parameters | Return Event |
| Get\_Att\_MTU\_Size | 0x5D | None | Status,  REPORT\_LE\_GATT\_EVENT (0x39) with the sub-event type Get\_Att\_MTU\_Size\_Res (0x05) |

**Command Parameters:**

None.

[[Return to Command Table]](#_UART_Command)

### DSP\_Runtime\_Program (0x30)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| DSP\_Runtime\_Program | 0x30 | Type, Cmd\_Buffer |  |

**Description:**

This command is used to change the EQ parameter of the DSP in run time.

**Precondition:**

EQ Mode should be enabled by using DSP configuration tool for type 0x13.

**Command Parameters:**

Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x13 | To Set Audio EQ parameter |
| 0x2D | To Set Sound Effect parameter |

Cmd Buffer: for Type 0x13 Length: 84 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Audio EQ Parameter  //SOFT  00000000c039631500000000c051eb85400000007fffffff18b42aef45232a27000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000276e41b8  //BASS  36b0f0758a53021235a323fc8b5e7fa738b3d27e875cb8cf3a61d55085aef0ef3ed11863812f30873f623e57809E0ae70000000000000000000000000000000000000000000000000000000000000000409a1c9f  //ROCK  2d35264093ba07232a70b4a7967869291997fe35b268d8f01ae7c4a9b1460ebb3916b5b986f9e32239a05624867055943eb27d61814dcb773f5672b280a9d684e648e12c13247096fada28801d6d1440623817ec |

Cmd Buffer: for Type 0x2D Length: 7 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Sound Effect Parameter  //MBC table  4f914469dd0d00 |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | For other than given Type and DSP Equalizer is not active |

[[Return to Command Table]](#_UART_Command)

### Read\_Vendor\_Eeprom\_Data (0x31)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Read\_Vendor\_Eeprom\_Data | 0x31 | Offset ,Length | [Report\_Vendor\_EEPROM\_Data](#_Report_Vendor_EEPROM_Data_(0x37)_1) |

**Description:**

This command is used to read the vendor EEPROM data.

**Precondition:**

None.

**Command Parameters:**

Offset: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Vendor EEPROM Offset |

Length: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Read Length |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x03 | Parameter error | Data offset + Data\_length is bigger than 32 (0x20) |

[[Return to Command Table]](#_UART_Command)

### Read\_IC\_Version\_Info (0x32)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Read\_IC\_Version\_Info | 0x32 | Reserved | [Report\_IC\_Ver\_Info](#_Report_IC_Ver_Info_(0x38)_1) |

**Description:**

This command is used to read the IC version information.

**Precondition:**

None.

**Command Parameters:**

Reserved: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reserved |

[[Return to Command Table]](#_UART_Command)

### Voice\_Prompt\_Ind\_Cmd (0x33)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Voice\_Prompt\_Ind\_Cmd | 0x33 | SubCommand Type,  SubCommand Payload |  |

**Description:**

This command is used to select voice prompt language (English/ Japanese/ Chinese) and predefined voice prompt source.

**Precondition:**

None.

**Command Parameters:**

SubCommand Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | voice prompt language selection |
| 0x01 | voice prompt source selection |

SubCommand\_Payload : for SubCommand 0x00 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | English |
| 0x01 | Japanese |
| 0x02 | Chinese |

SubCommand\_Payload : for SubCommand 0x01 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | fully charge |
| 0x01 | battery about 75% |
| 0x02 | battery about 50% |
| 0x03 | battery about 20% |
| 0x04 | please charge |
| 0x05 | left |
| 0x06 | right |
| 0x07 | connected |
| 0x08 | pairing mode |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x03 | Parameter error | Incorrect sub command parameter |

[[Return to Command Table]](#_UART_Command)

### Read\_BTM\_Link\_Mode (0x34)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Read\_BTM\_Link\_Mode | 0x34 | Reserved | [Report\_BTM\_Link\_Mode](#_Report_BTM_Link_Mode_(0x3A)_1) |

**Description:**

This command is used to read the last link mode, it can be used to identify the mode before give the Porfile\_Link\_Back command.

**Precondition:**

None.

**Command Parameters:**

Reserved: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reserved |

[[Return to Command Table]](#_UART_Command)

### Configure\_Vendor\_Parameter (0x35)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Configure\_Vendor\_Parameter | 0x35 | Opcode, (Mandatory)  Option, (Mandatory)  Length, (Optional)  Parameters (Optional) |  |

**Description:**

This command is used to configure the vendor parameter.

**Precondition:**

None.

**Command Parameters:**

Opcode: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | **Change Device Name:** The device name in EEPROM will be replaced by new device name parameter. |
| Others | Reserved |

Option: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Reserved, Set it to 0x00 |

Length: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Length of the following parameter |

Parameter: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | New device name parameter |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | Length is zero or bigger than 64 when use Restore Device Name |
| 0x03 | Parameter error | Length is bigger than 23 when use Change Device Name  OpCode incorrect |

[[Return to Command Table]](#_UART_Command)

### nSPK Exchange\_Link\_Info\_Cmd (0x37)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| nSPK Exchange\_Link\_Info\_Cmd | 0x37 | Exchange\_data |  |

**Description:**

It is for information exchange when creating CSB link. This command must be sent before creating CSB link if it has data need to exchange.

**Precondition:**

None.

**Command Parameters:**

Exchange\_data: Length: 16 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX… | 16 bytes exchanged data. |

[[Return to Command Table]](#_Report_Read_Feature_List_Reply_(0x4)



### [nSPK Set GIAC(0x38)](#_Report_Read_Feature_List_Reply_(0x4)

|  |  |  |  |
| --- | --- | --- | --- |
| [Command](#_Report_Read_Feature_List_Reply_(0x4) | [Op Code](#_Report_Read_Feature_List_Reply_(0x4) | [Command Parameters](#_Report_Read_Feature_List_Reply_(0x4) | [Return Event](#_Report_Read_Feature_List_Reply_(0x4) |
| [nSPK Set GIAC](#_Report_Read_Feature_List_Reply_(0x4) | [0x38](#_Report_Read_Feature_List_Reply_(0x4) | [GIAC](#_Report_Read_Feature_List_Reply_(0x4) |  |

**[Description:](#_Report_Read_Feature_List_Reply_(0x4)**

[This command is used for change the group code for CSB general pairing and the modification won’t change the EEPROM setting.](#_Report_Read_Feature_List_Reply_(0x4)

**[Precondition:](#_Report_Read_Feature_List_Reply_(0x4)**

[None.](#_Report_Read_Feature_List_Reply_(0x4)

**[Command Parameters:](#_Report_Read_Feature_List_Reply_(0x4)**

[GIAC: Length: 2 Byte](#_Report_Read_Feature_List_Reply_(0x4)

|  |  |
| --- | --- |
| [Value](#_Report_Read_Feature_List_Reply_(0x4) | [Parameter Description](#_Report_Read_Feature_List_Reply_(0x4) |
| [0xXXXX…](#_Report_Read_Feature_List_Reply_(0x4) | [GIAC value](#_Report_Read_Feature_List_Reply_(0x4) |

[[Return to Command Table]](#_UART_Command)

### [READ\_FEATURE\_LIST (0x39)](#_Report_Read_Feature_List_Reply_(0x4)

|  |  |  |  |
| --- | --- | --- | --- |
| [Command](#_Report_Read_Feature_List_Reply_(0x4) | [Op Code](#_Report_Read_Feature_List_Reply_(0x4) | [Command Parameters](#_Report_Read_Feature_List_Reply_(0x4) | [Return Event](#_Report_Read_Feature_List_Reply_(0x4) |
| [READ\_FEATURE\_LIST](#_Report_Read_Feature_List_Reply_(0x4) | [0x39](#_Report_Read_Feature_List_Reply_(0x4) | [Reserved](#_Report_Read_Feature_List_Reply_(0x4) | [Report\_Read\_Feature\_List\_Reply](#_Report_Read_Feature_List_Reply_(0x4) |

**[Description:](#_Report_Read_Feature_List_Reply_(0x4)**

[This command is used to query supported feature.](#_Report_Read_Feature_List_Reply_(0x4)

**[Precondition:](#_Report_Read_Feature_List_Reply_(0x4)**

[None.](#_Report_Read_Feature_List_Reply_(0x4)

**[Command Parameters:](#_Report_Read_Feature_List_Reply_(0x4)**

[Reserved: Length: 1 Byte](#_Report_Read_Feature_List_Reply_(0x4)

|  |  |
| --- | --- |
| [Value](#_Report_Read_Feature_List_Reply_(0x4) | [Parameter Description](#_Report_Read_Feature_List_Reply_(0x4) |
| [0xXX](#_Report_Read_Feature_List_Reply_(0x4) | [Reserved](#_Report_Read_Feature_List_Reply_(0x4) |

[[Return to Command Table]](#_UART_Command)

### Personal\_MSPK\_GROUP\_Control (0x3A)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| Personal\_MSPK\_GROUP\_Control | 0x3A | Enable, Master\_SPK\_BD\_ADDR |  |

**Description:**

This command is used to inform Slave\_SPK for Personal MSPK Group Setting enable or disable.

The Master\_SPK\_BD\_ADDR is Master\_SPK BT Device Address. Slave\_SPK will only accept the MSPK connection for specific Master\_SPK if Personal MSPK Group is enabled.

This command must be assert before MSPK Command triggered.

**Precondition:**

None.

**Command Parameters:**

Enable: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Disable this feature |
| 0x01 | Enable this feature |

Master\_SPK\_BD\_ADDR: Length: 6 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXXXXXXXXXX | The BT Device Address of Master\_SPK. It is Big endian.  EX: BT Device Address is “00:11:22:33:44:55”.  Then it should be represented as “0x001122334455”. |

[[Return to Command Table]](#_UART_Command)

### UART\_CMD\_TEST\_DEVICE (0x3B)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| UART\_CMD\_TEST\_DEVICE | 0x3B | Test OP code |  |

**Description:**

This command is used to test the assigned device.

**Precondition:**

None.

**Command Parameters:**

Test OP code: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Test CP chip |
| 0x01~0xFF | Reserved |

[[Return to Command Table]](#_UART_Command)

# MMI\_Action UART Command (0x02)

|  |  |  |  |
| --- | --- | --- | --- |
| Command | Op Code | Command Parameters | Return Event |
| MMI\_Action | 0x02 | data\_base\_index, action |  |

**Description:**

MCU can send proper command to complete different kinds of action.

**Command Parameters:**

data\_base\_index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xNN | database index of dedicate HF device bit0~2: data base index bit3~5: RFCOMM index if 2 SPP feature is enabled |

#### Action:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Action | | Support version | | |
| Value | **Parameter Description** | **Start** | **End** | |
| [0x01](#_Action_0x01:) | add/remove SCO link | V2.00 | |  | |
| 0x02 | force end active call |  | |  | |
| [0x04](#_Action_0x04:) | Accept an incoming call | V2.00 | |  | |
| [0x05](#_Action_0x05:) | Reject an incoming call | V2.00 | |  | |
| [0x06](#_Action_0x06:) | End call/ transfer audio to phone | V2.00 | |  | |
| [0x07](#_Action_0x07:) | Toggle microphone on/off | V2.00 | |  | |
| [0x08](#_Action_0x08:) | Mute microphone | V2.00 | |  | |
| [0x09](#_Action_0x09:) | Active microphone | V2.00 | |  | |
| [0x0A](#_Action_0x0A:) | voice dial | V2.00 | |  | |
| [0x0B](#_Action_0x0B:) | cancel voice dial | V2.00 | |  | |
| [0x0C](#_Action_0x0C:) | last number redial | V2.00 | |  | |
| [0x0D](#_Action_0x0D:) | Set the active call on hold and active the hold call | V2.00 | |  | |
| [0x0E](#_Action_0x0E:) | switch voice between phone and headset | V2.00 | |  | |
| [0x0F](#_Action_0x0F:) | Query call list information(CLCC) | V2.00 | |  | |
| [0x10](#_Action_0x10:) | three way call | V2.00 | |  | |
| [0x11](#_Action_0x11:) | release the waiting call or on hold call | V2.00 | |  | |
| [0x12](#_Action_0x12:) | accept the waiting call or active the on hold call and release the active call | V2.00 | |  | |
| 0x16 | initiate HF connection |  | |  | |
| [0x17](#_Action_0x17:) | disconnect HF link | V2.00 | |  | |
| 0x18 | Enable RX noise reduction when SCO ready |  | |  | |
| 0x19 | Disable RX noise reduction when SCO ready |  | |  | |
| 0x1A | Switch RX noise reduction when SCO ready |  | |  | |
| 0x1B | Enable TX noise reduction when SCO ready |  | |  | |
| 0x1C | Disable TX noise reduction when SCO ready |  | |  | |
| 0x1D | Switch TX noise reduction when SCO ready |  | |  | |
| [0x1E](#_Action_0x1E:) | Enable AEC when SCO ready | V2.00 | |  | |
| [0x1F](#_Action_0x1F:) | Disable AEC when SCO ready | V2.00 | |  | |
| [0x20](#_Action_0x20:) | Switch AEC enable/disable when SCO ready | V2.00 | |  | |
| [0x21](#_Action_0x21:) | Enable AEC RX noise reduction when SCO ready | V2.00 | |  | |
| [0x22](#_Action_0x22:) | Disable AEC RX noise reduction when SCO ready | V2.00 | |  | |
| [0x23](#_Action_0x23:) | Switch AEC RX noise reduction when SCO ready | V2.00 | |  | |
| [0x24](#_Action_0x24:) | increase microphone gain | V2.00 | |  | |
| [0x25](#_Action_0x25:) | decrease microphone gain | V2.00 | |  | |
| [0x26](#_Action_0x26:) | switch primary HF device and secondary HF device role | V2.00 | |  | |
| 0x30 | increase speaker gain |  | |  | |
| 0x31 | decrease speaker gain |  | |  | |
| 0x32 | Play/Pause music |  | |  | |
| 0x33 | Stop music |  | |  | |
| 0x34 | Next song |  | |  | |
| 0x35 | Previous song |  | |  | |
| 0x36 | Fast forward |  | |  | |
| 0x37 | Rewind |  | |  | |
| 0x38 | EQ mode up |  | |  | |
| 0x39 | EQ mode down |  | |  | |
| [0x3A](#_Action_0x3A:) | Lock button | V2.00 | |  | |
| [0x3B](#_Action_0x3B:) | Disconnect A2DP link | V2.00 | |  | |
| [0x3C](#_Action_0x3C:) | next audio effect | V2.00 | |  | |
| [0x3D](#_Action_0x3D:) | previous audio effect | V2.00 | |  | |
| 0x3E | Toggle 3D effect |  | |  | |
| [0x3F](#_Action_0x3F:) | report current EQ mode | V2.00 | |  | |
| [0x40](#_Action_0x40:) | report current audio effect status | V2.00 | |  | |
| 0x50 | enter pairing mode (from power off state) |  | |  | |
| [0x51](#_Action_0x51:) | power on button press | V2.00 | |  | |
| [0x52](#_Action_0x52:) | power on button release | V2.00 | |  | |
| [0x53](#_Action_0x53:) | power off button press | V2.00 | |  | |
| [0x54](#_Action_0x54:) | power off button release | V2.00 | |  | |
| [0x55](#_Action_0x55:) | Reverse panel | V2.00 | |  | |
| [0x56](#_Action_0x56:) | Reset some EEPROM setting to default setting | V2.00 | |  | |
| [0x57](#_Action_0x57:) | Force speaker gain toggle | V2.00 | |  | |
| [0x58](#_Action_0x58:) | Toggle button indication | V2.00 | |  | |
| [0x59](#_Action_0x59~0x5C:) | Combine function 0 | V2.00 | |  | |
| [0x5A](#_Action_0x59~0x5C:) | Combine function 1 | V2.00 | |  | |
| [0x5B](#_Action_0x59~0x5C:) | Combine function 2 | V2.00 | |  | |
| [0x5C](#_Action_0x59~0x5C:) | Combine function 3 | V2.00 | |  | |
| [0x5D](#_Action_0x5D:) | fast enter pairing mode (from non-off mode) | V2.00 | |  | |
| [0x5E](#_Action_0x5E:) | Switch power OFF | V2.00 | |  | |
| [0x5F](#_Action_0x5F:) | Disable LED | V2.00 | |  | |
| [0x60](#_Action_0x60:) | Toggle buzzer | V2.00 | |  | |
| [0x61](#_Action_0x61:) | Disable buzzer | V2.00 | |  | |
| [0x62](#_Action_0x62:) | Enable buzzer | V2.00 | |  | |
| [0x63](#_Action_0x63:) | Change tone set (SPK module support two sets of tone) | V2.00 | |  | |
| [0x64](#_Action_0x64_~) | Retrieve phonebook |  | |  | |
| [0x65](#_Action_0x64_~) | Retrieve MCH |  | |  | |
| [0x66](#_Action_0x64_~) | Retrieve ICH |  | |  | |
| [0x67](#_Action_0x64_~) | Retrieve OCH |  | |  | |
| [0x68](#_Action_0x64_~) | Retrieve CCH |  | |  | |
| [0x69](#_Action_0x64_~) | Cancel access PBAP |  | |  | |
| [0x6A](#_Action_0x6A:) | Indicate battery status | V2.00 | |  | |
| [0x6B](#_Action_0x6B:) | Exit pairing mode | V2.00 | |  | |
| 0x6C | link last device |  | |  | |
| 0x6D | disconnect all link |  | |  | |
| 0x6E | OHS event 1 |  | |  | |
| 0x6F | OHS event 2 |  | |  | |
| 0x70 | OHS event 3 |  | |  | |
| 0x71 | OHS event 4 |  | |  | |
| [0x72](#_Action_0x72~0x76:) | SHS\_SEND\_USER\_DATA\_1 (for embedded application mode) | V2.00 | |  | |
| [0x73](#_Action_0x72~0x76:) | SHS\_SEND\_USER\_DATA\_2 (for embedded application mode) | V2.00 | |  | |
| [0x74](#_Action_0x72~0x76:) | SHS\_SEND\_USER\_DATA\_3 (for embedded application mode) | V2.00 | |  | |
| [0x75](#_Action_0x72~0x76:) | SHS\_SEND\_USER\_DATA\_4 (for embedded application mode) | V2.00 | |  | |
| [0x76](#_Action_0x72~0x76:) | SHS\_SEND\_USER\_DATA\_5 (for embedded application mode) | V2.00 | |  | |
| [0x77](#_Action_0x77:) | report current RX NR status | V2.00 | |  | |
| [0x78](#_Action_0x78:) | report current TX NR status | V2.00 | |  | |
| 0x79 | force buzzer alarm |  | |  | |
| [0x7A](#_Action_0x7A:) | Cancel all BT paging | V2.00 | |  | |
| 0x7B | OHS event 5 |  | |  | |
| 0x7C | OHS event 6 |  | |  | |
| [0x7D](#_Action_0x7D:) | Disconnect SPP link | V2.00 | |  | |
| 0xC0 | OHS event 7 (for Unlimiter project) |  | |  | |
| 0xC1 | OHS event 8 (for Unlimiter project) |  | |  | |
| 0xC2 | OHS event 9 (for Unlimiter project) |  | |  | |
| 0xC3 | OHS event 10 (for Unlimiter project) |  | |  | |
| 0xC4 | OHS event 11 (for Unlimiter project) |  | |  | |
| 0xC5 | OHS event 12 (for Unlimiter project) |  | |  | |
| 0xC6 | OHS event 13 (for Unlimiter project) |  | |  | |
| 0xC7 | OHS event 14 (for Unlimiter project) |  | |  | |
| 0xC8 | OHS event 15 (for Unlimiter project) |  | |  | |
| 0xC9 | OHS event 16 (for Unlimiter project) |  | |  | |
| 0xCA | OHS event 17 (for Unlimiter project) |  | |  | |
| 0xCB | Switch DSP mode 2 or 3 (for Unlimiter project) |  | |  | |
| [0xE0](#_Action_0xE0:) | Trigger NSPK Master | V2.00 | |  | |
| [0xE1](#_Action_0xE1:) | Trigger NSPK Slave | V2.00 | |  | |
| [0xE2](#_Action_0xE2:) | NSPK one key connect/disconnect | V2.00 | |  | |
| [0xE3](#_Action_0xE3:) | Cancel NSPK creation | V2.00 | |  | |
| [0xE4](#_Action_0xE4:) | Terminate NSPK link | V2.00 | |  | |
| [0xE5](#_Action_0xE5:) | Terminate / Cancel NSPK connection | V2.00 | |  | |
| [0xE6](#_Action_0xE6:) | NSPK Master enter Aux-in 44.1K PCM Encoder mode | V2.00 | |  | |
| [0xE7](#_Action_0xE7:) | NSPK Master enter Aux-in 48K PCM Encoder mode | V2.00 | |  | |
| [0xE8](#_Action_0xE8:) | NSPK Master exit Aux-in PCM Encoder mode | V2.00 | |  | |
| [0xE9](#_Action_0xE9:) | NSPK Master enter Aux-in SBC Encoder mode | V2.00 | |  | |
| 0xEA | NSPK Master exit Aux-in SBC Encoder mode |  | |  | |
| [0xEB](#_Action_0xEB:) | NSPK dynamic creation | V2.00 | |  | |
| [0xEC](#_Action_0xEC:) | NSPK switch channel | V2.00 | |  | |
| [0xED](#_Action_0xED:) | NSPK power off all speakers | V2.00 | |  | |
| [0xEE](#_Action_0xEE:) | NSPK AFH SBCENCODING AUDIOSYNC | V2.00 | |  | |
| 0xF0 | NSPK MASTER PAGE SLAVE FOR NEW SLAVE |  | |  | |
| 0xF1 | NSPK SLAVE ENABLE PAGE SCAN FOR NEW MASTER |  | |  | |
| 0xF2 | NSPK SLAVE USE SLOW PAGE SCAN |  | |  | |
| 0xF3 | NSPK SLAVE USE FAST PAGE SCAN |  | |  | |
| [0xF4](#_Action_0xF4:) | NSPK\_ENTER\_NSPK\_MODE | V2.00 | |  | |
| [0xF5](#_Action_0xF5:) | NSPK\_ENTER\_BROADCAST\_MODE | V2.00 | |  | |
| [0xF6](#_Action_0xF6:) | NSPK\_ADD\_THIRD\_SPK | V2.00 | |  | |
| [0xF7](#_Action_0xF7) | NSPK\_SOUND\_SYNCHRONIZATION | V2.00 | |  | |
| [0xF8](#_Action_0xF8) | NSPK\_CSB\_CONNECTED\_MODE\_SWITCH | V2.00 | |  | |
| [0xF9](#_Action_0xF9) | NSPK back to last mode | V2.00 | |  | |

#### Action 0x01:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x01 | add/remove SCO link |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection |

Description:

This action is used to create a SCO link if the ACL link exists and no SCO connection.

And it also can disconnect the SCO link if the SCO link exists.

[[Return to Action Table]](#_Action:)

#### Action 0x04:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x04 | Accept an incoming call |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection  No incoming call on going |

Description:

This action is used to answer the incoming call.

[[Return to Action Table]](#_Action:)

#### Action 0x05:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x05 | Reject an incoming call |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection  No incoming call on going |

Description:

This action is used to reject the incoming call.

[[Return to Action Table]](#_Action:)

#### Action 0x06:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x06 | 1. End call if SCO exist.  2. Voice transfer to headset if SCO not exist. |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection  No Call active  No outgoing call on going |

Description:

This action is used to hang-up the active call if outgoing call ongoing or SCO exist.

It also can transfer the voice to headset if SCO not exist.

If device supports 3-way call and there is on hold or wait call, this action is able to disconnect all the active call and keep the on hold or wait call.

[[Return to Action Table]](#_Action:)

#### Action 0x07:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x07 | 1. Mute microphone if microphone is not mute  2. Active microphone if microphone is mute |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection  No Call active  No SCO connection |

Description:

This action is used to mute or active microphone while CALL active or SCO exists.

[[Return to Action Table]](#_Action:)

#### Action 0x08:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x08 | Mute microphone |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection  No Call active  No outgoing call on going |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection  No Call active  No SCO connection  Microphone is mute |

Description:

This action is used to mute microphone while CALL active or SCO exists.

[[Return to Action Table]](#_Action:)

#### Action 0x09:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x09 | Un-mute microphone |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection  No Call active  No SCO connection  Microphone is unmute |

Description:

This action is used to unmute microphone while CALL active or SCO exists.

[[Return to Action Table]](#_Action:)

#### Action 0x0A:

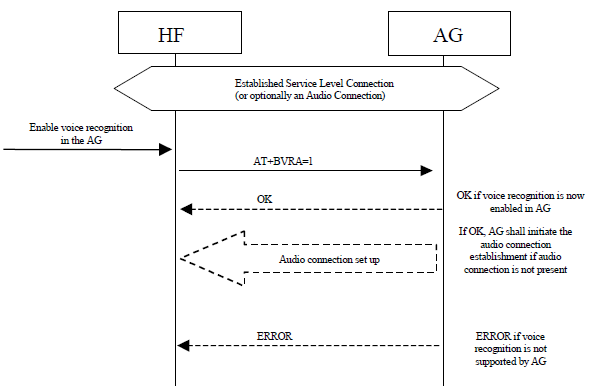
|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x0A | Voice dial |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | Not support Voice recognition  Already has active call. |

Description:

This action is used to activate voice dial if HF is connected. If there is no HF connection, it initiates the HF connection and activate the voice recognition application (ex: Siri)



[[Return to Action Table]](#_Action:)

#### Action 0x0B:

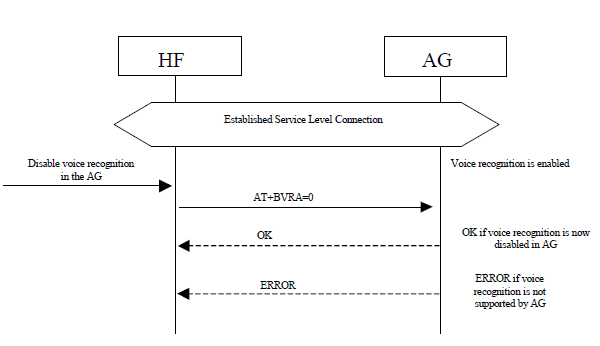
|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x0B | Cancel voice dial |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection  No Voice call  No activation Voice call ongoing. |

Description:

This action is used to deactivate voice dial if HF is connected.



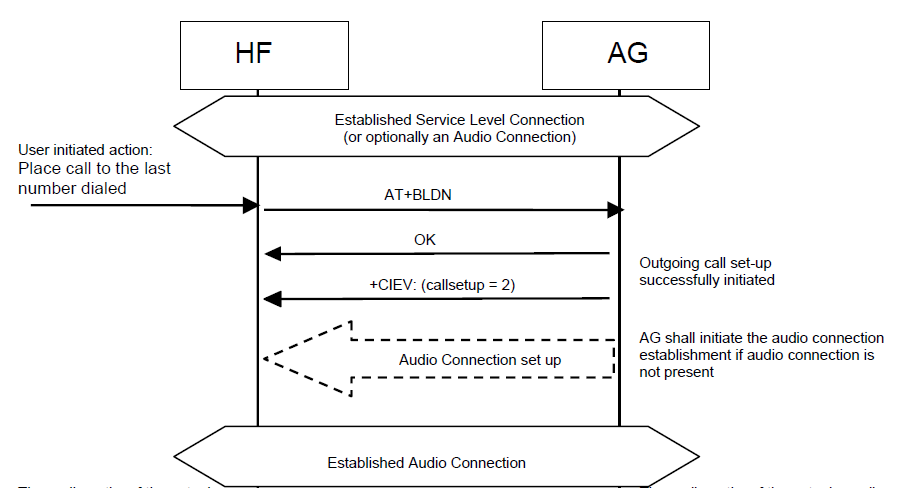
[[Return to Action Table]](#_Action:)

#### Action 0x0C:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x0C | Last number redial |

Description:

This action is used to redial the last number if HF is connected. If there is no HF connection, it initiates the HF connection first and active last number redial.



[[Return to Action Table]](#_Action:)

#### Action 0x0D:

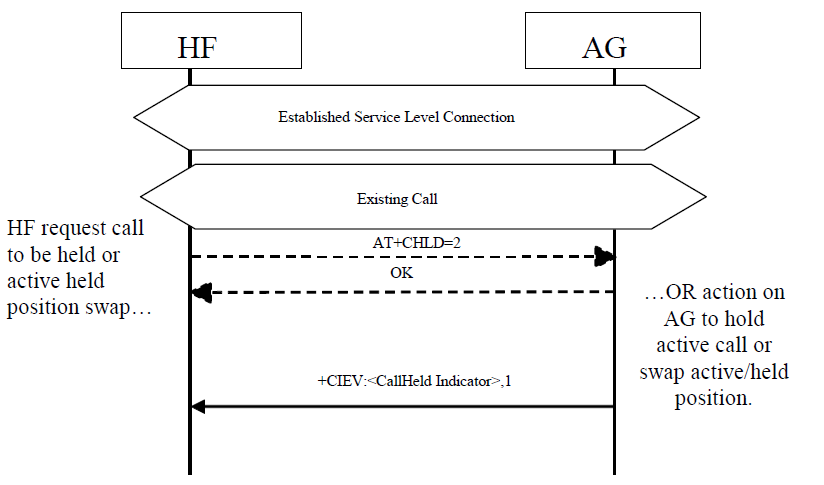
|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x0D | Set the active call on hold and active the hold call |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection  Not support 3-way calling  No active call while Call on hold and outgoing call ongoing. |

Description:

This action is used to swap the active/held call position. If cell phone doesn’t support 3-way calling, it hangs up the call.



[[Return to Action Table]](#_Action:)

#### Action 0x0E:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x0E | voice transfer |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No SCO connection existing.  No active call.  No outgoing call ongoing. |

Description:

This action is used to switch voice between phone and headset. If HF isn’t connected, SPK create the HF connection.

[[Return to Action Table]](#_Action:)

#### Action 0x0F:

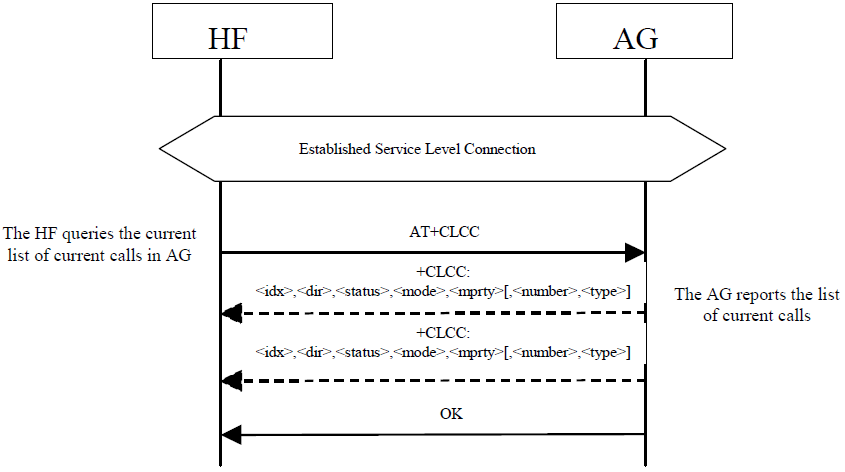
|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x0F | trigger SPK Module to query call list information |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection |

Description:

The HF shall execute this procedure to query the list of current calls in AG. Refer to section 7.18[[1]](#_References) and section 4.31.1[[3]](#_References).



[+CLCC:<id1>,<dir>,<stat>,<mode>,<mpty>[,<number>,<type>[,<alpha>]][<CR><LF>+CLCC:<id2>,<dir>,<stat>,<mode>,<mpty>[,<number>,<type>[,<alpha>]][...]]]

|  |
| --- |
| **<idx>** Call identification number |
| **<dir>** |
| 0:Mobile originated (MO) call |
| 1:Mobile originated (MT) call |
| **<state>** State of the call |
| 0:ACTIVE |
| 1:HELD |
| 2: Dialing (MO) |
| 3: Alerting (MO) |
| 4: Incoming (MT) |
| 5: Waiting (MT) |
| **<mode>** |
| 0: Voice |
| 1: Data |
| 2: Fax |
| 3: Voice followed by data, voice mode |
| 4: Alternating voice/data, voice mode |
| 5: Alternating voice/fax, voice mode |
| 6: Voice followed by data, data mode |
| 7: Alternating voice/data, data mode |
| 8: Alternating voice/fax, fax mode |
| 9: Unknown |
| **<mpty>** |
| 0:Call is not one of multiparty (conference) call parties |
| 1:Call is one of multiparty (conference) call parties |

[[Return to Action Table]](#_Action:)

#### Action 0x10:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x10 | Three way call |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection |

Description:

This action is used to add a held call into the conference if the device/operator supports 3 way call feature.

[[Return to Action Table]](#_Action:)

#### Action 0x11:

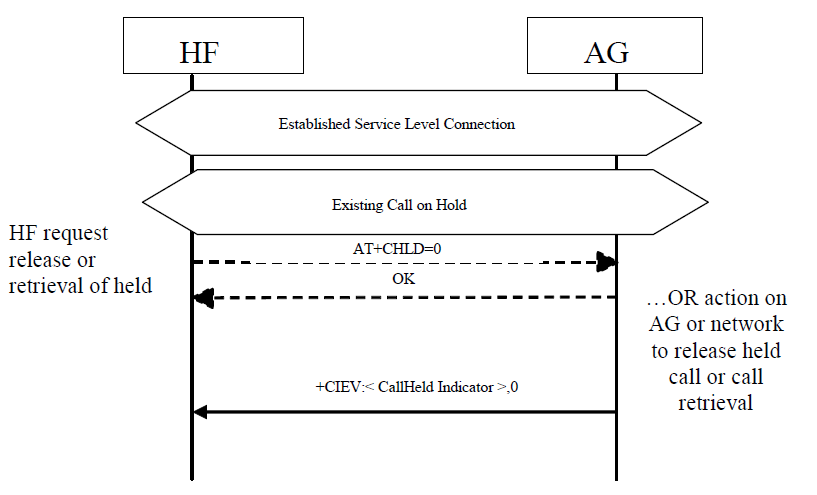
|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x11 | Release the waiting call or on hold call |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection |

Description:

This action is used to release all held call.



[[Return to Action Table]](#_Action:)

#### Action 0x12:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x12 | Accept the waiting call or active the on hold call and release the active call |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection |

Description:

This action is used to release active call and accept the waiting/held call if the device supports 3 way calls.

[[Return to Action Table]](#_Action:)

#### Action 0x17:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x17 | disconnect HF link |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection  No ACL connection and no HF/HS connected profile |

Description:

This action is used to disconnect HF connection.

[[Return to Action Table]](#_Action:)

#### ~~Action 0x1E:~~

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x1E | Enable AEC when SCO ready |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection  No SCO connection and DSP not support AEC |

Description:

This action is used to enable AEC.

[[Return to Action Table]](#_Action:)

#### ~~Action 0x1F:~~

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x1F | Disable AEC when SCO ready |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection  No SCO connection and DSP not support AEC |

Description:

This action is used to disable AEC.

[[Return to Action Table]](#_Action:)

#### ~~Action 0x20:~~

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x20 | Switch AEC when SCO ready |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection  No SCO connection and DSP not support AEC |

Description:

This action is used to switch AEC.

[[Return to Action Table]](#_Action:)

#### ~~Action 0x21:~~

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x21 | Enable AEC and RX noise reduction when SCO ready |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection  No SCO connection and DSP not support AEC |

Description:

This action is used to enable AEC and RX noise reduction.

[[Return to Action Table]](#_Action:)

#### ~~Action 0x22:~~

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x22 | Disable AEC and RX noise reduction when SCO ready |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection  No SCO connection and DSP not support AEC/NR |

Description:

This action is used to disable AEC and RX noise reduction.

[[Return to Action Table]](#_Action:)

#### ~~Action 0x23:~~

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x23 | Switch AEC and RX noise reduction when SCO ready |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection  No SCO connection and DSP not support AEC/NR |

Description:

This action is used to switch AEC and RX noise reduction.

[[Return to Action Table]](#_Action:)

#### Action 0x24:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x24 | Volume up the microphone gain |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection |

Description:

This action is used to volume up the microphone gain.

[[Return to Action Table]](#_Action:)

#### Action 0x25:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x25 | Volume down the microphone gain |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection |

Description:

This action is used to volume down the microphone gain.

[[Return to Action Table]](#_Action:)

#### Action 0x26:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x26 | switch primary HF device and secondary HF device role |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No HF connection |

Description:

This action is used to switch the primary/secondary HF device role when SPK supports multiple connections. .

[[Return to Action Table]](#_Action:)

#### Action 0x3A:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x3A | Lock button |

Description:

This action is used to lock system.

[[Return to Action Table]](#_Action:)

#### Action 0x3B:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x3B | Disconnect A2DP link |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No ACL connection  No A2DP connected |

Description:

This action is used to disconnect A2DP connection.

[[Return to Action Table]](#_Action:)

#### Action 0x3C:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x3C | Next audio effect |

Description:

This action is used to switch to next audio effect.

[[Return to Action Table]](#_Action:)

#### Action 0x3D:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x3D | Previous audio effect |

Description:

This action is used to switch to previous audio effect.

[[Return to Action Table]](#_Action:)

#### Action 0x3F:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x3F | Report current EQ mode |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | DSP doesn’t support EQ mode |

Description:

This action is used to report current EQ mode. If embedded mode enable, current EQ status report via SPP connection.

[[Return to Action Table]](#_Action:)

#### Action 0x40:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x40 | Report current audio effect status |

Description:

This action is used to report current audio effect status. If embedded mode enable, current audio effect status report via SPP connection.

[[Return to Action Table]](#_Action:)

#### Action 0x51:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x51 | Power on button press |

Description:

This action is used to press power on button.

[[Return to Action Table]](#_Action:)

#### Action 0x52:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x52 | Power on button release |

Description:

This action is used to release power on button.

[[Return to Action Table]](#_Action:)

#### Action 0x53:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x53 | Power off button press |

Description:

This action is used to press power off button.

[[Return to Action Table]](#_Action:)

#### Action 0x54:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x54 | Power off button release |

Description:

This action is used to release power off button.

[[Return to Action Table]](#_Action:)

#### Action 0x55:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x55 | Reverse panel |

Description:

This action is used to release reverse panel.

[[Return to Action Table]](#_Action:)

#### Action 0x56:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x56 | Reset some EEPROM setting to default value |

Description:

This action is used to reset some EEPROM setting to default value.

|  |  |
| --- | --- |
| Device\_List\_Table | Erase the device table: 0x00 |
| linked\_priority | Erase the linked priority: 0x00 |
| mic\_gain\_level | Microphone gain: 0XDD |
| device\_speaker\_gain | Speaker gain : 0x99 |
| device\_absolute\_volume | Absolute volume: 0x99 |
| app\_function\_status | buzzer status : OFF  BT\_STABDBY\_MODE\_BIT: keep the speaker connectable in power off state. disabled  VOL\_CTRL\_DIRECT\_BIT: circular volume control direction.  0: volume up when volume control |
| hf\_device\_index | Record HF primary device index: 0xFF |
| a2dp\_device\_index | Record A2DP device: 0xFF |
| line\_in\_gain\_level | Line-in gain: 0x99 |
| CSB\_DIAC | The access code for dedicate CSB pairing: 0x00000000 |
| CSB\_Trigger\_Sync\_Role\_Setting | Sync role setting: 0x00 |
| NSPK\_channel | nSPK channel: 0x21 |

[[Return to Command Table]](#_MMI_Action_UART_Command)

#### Action 0x57:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x57 | Force speaker gain toggle |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No Call exist |

Description:

This action is used to release power off button.

[[Return to Action Table]](#_Action:)

#### Action 0x58:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x58 | Toggle button indication |

Description:

This action is used to toggle button indication depends on the setting of EEPROM output\_indication\_table.

output\_indication\_table[0] : indication GPIO0

output\_indication\_table[1] : indication GPIO1

[[Return to Action Table]](#_Action:)

#### Action 0x59~0x5C:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x59 | Combine function 0 |
| 0x5A | Combine function 1 |
| 0x5B | Combine function 2 |
| 0x5C | Combine function 3 |

Description:

Those 4 actions are used to trigger combine function 0~3.

[[Return to Action Table]](#_Action:)

#### Action 0x5D:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x5D | fast enter pairing mode (from non-off mode) |

data\_base\_index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | BTM generate tone when enter pairing mode |
| Others | BTM doesn't generate tone when enter pairing mode |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | SPK is in OFF state |

Description:

This action is used to enter pairing mode from non-off mode.

[[Return to Action Table]](#_Action:)

#### Action 0x5E:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x5E | switch power off |

Description:

This action is used to execute the power off process. If nSPK is connected, this action is used to disconnect the remote nSPK first then MCU receive EVENT [BTM\_Utility\_Rsp](#_BTM_Utility_Req_(0x1B)) (0x1B) with action\_type 0x05.

After receive the event, MCU can send MMI\_Action [0x53](#_Action_0x53:), [0x54](#_Action_0x54:) to power off current SPK.

If it’s a standalone SPK, this action executes the power off directly.

[[Return to Action Table]](#_Action:)

#### Action 0x5F:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x5F | Disable LED |

Description:

This action is used to disable LED.

[[Return to Action Table]](#_Action:)

#### Action 0x60:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x60 | Toggle buzzer |

Description:

This action is used to toggle buzzer if Buzzer function implement.

[[Return to Action Table]](#_Action:)

#### Action 0x61:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x61 | Disable buzzer |

Description:

This action is used to disable buzzer if Buzzer function implement.

[[Return to Action Table]](#_Action:)

#### Action 0x62:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x62 | Enable buzzer |

Description:

This action is used to enable buzzer if Buzzer function implement.

[[Return to Action Table]](#_Action:)

#### Action 0x63:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x63 | Switch to next support ringtone language set. |

Description:

This action is used to switch to next support ringtone language set.

[[Return to Action Table]](#_Action:)

#### Action 0x64 ~ 0x69:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x64 | Retrieve phonebook |
| 0x65 | Retrieve MCH |
| 0x66 | Retrieve ICH |
| 0x67 | Retrieve OCH |
| 0x68 | Retrieve CCH |
| 0x69 | Cancel access PBAP |

Description:

This action is used to access phone book if it supports PBAP. Those actions only support in some firmware.

[[Return to Action Table]](#_Action:)

#### Action 0x6A:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x6A | Indicate battery status |

Description:

This action is used to indicate battery (high, medium, low) status via voice prompt if device support voice prompt report battery status feature or LED indication.

[[Return to Action Table]](#_Action:)

#### Action 0x6B:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x6B | Exit pairing mode |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | SPK is in paring mode/ SPP connection existing  SPK has A2DP connection/HF HS connection if device support social mode. |

Description:

This action is used to exit pairing mode.

[[Return to Action Table]](#_Action:)

#### Action 0x72~0x76:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x72 | SHS\_SEND\_USER\_DATA\_1 (for embedded application mode) |
| 0x73 | SHS\_SEND\_USER\_DATA\_2 (for embedded application mode) |
| 0x74 | SHS\_SEND\_USER\_DATA\_3 (for embedded application mode) |
| 0x75 | SHS\_SEND\_USER\_DATA\_4 (for embedded application mode) |
| 0x76 | SHS\_SEND\_USER\_DATA\_5 (for embedded application mode) |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | Embedded mode disable |

Description:

Those 5 actions are used to handle user MMI data if device support Embedded SPP function.

[[Return to Action Table]](#_Action:)

#### Action 0x77:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x77 | Report current RX NR status |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | DSP not support NR mode |

Description:

This action is used to report current RX NR status.

[[Return to Action Table]](#_Action:)

#### Action 0x78:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x78 | Report current TX NR status |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | DSP not support NR mode |

Description:

This action is used to report current TX NR status.

[[Return to Action Table]](#_Action:)

#### Action 0x7A:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x7A | Cancel all BT paging |

Description:

This action is used to cancel BT paging.

[[Return to Action Table]](#_Action:)

#### Action 0x7D:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x7D | Disconnect SPP link |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | No ACL connection  No SPP profile connected |

Description:

This action is used to disconnect SPP link.

[[Return to Action Table]](#_Action:)

#### Action 0xE0:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xE0 | Trigger NSPK Master |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | Call active  Line-in/Aux-in CSB mode disable |

Description:

This action is used to enter NSPK Master mode.

[[Return to Action Table]](#_Action:)

#### Action 0xE1:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xE1 | Trigger NSPK Slave |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | Call active  Line-in/Aux-in CSB mode disable |

Description:

This action is used to enter NSPK Slave mode.

[[Return to Action Table]](#_Action:)

#### Action 0xE2:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xE2 | NSPK one key connect/disconnect |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | Call active  Line-in/Aux-in CSB mode disable  CSB state doesn’t under “Connecting”, “Connected” or “Add 3rd SPK” |

Description:

This action is used for NSPK one key connect/disconnect.

[[Return to Action Table]](#_Action:)

#### Action 0xE3:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xE3 | Cancel NSPK create connection |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | CSB state doesn’t under “Connecting”, “Connected” or “Add 3rd SPK” |

Description:

This action is used to cancel NSPK create connection.

[[Return to Action Table]](#_Action:)

#### Action 0xE4:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xE4 | Terminate NSPK link |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | CSB state doesn’t under “Connected” or “Add 3rd SPK” |

Description:

This action is used to terminate NSPK link.

[[Return to Action Table]](#_Action:)

#### Action 0xE5:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xE5 | Terminate / Cancel NSPK connection |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | CSB state doesn’t under “Connecting”, “Connected”, “Wait to create CSB link” or “Add 3rd SPK” |

Description:

This action is used to terminate/cancel NSPK connection.

[[Return to Action Table]](#_Action:)

#### Action 0xE6:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xE6 | NSPK Master enter Aux-in 44.1K PCM Encoder mode |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | SPK is slave role |

Description:

This action is used for NSPK master enter AUX-IN 44.1K PCM encoder mode.

[[Return to Action Table]](#_Action:)

#### Action 0xE7:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xE7 | NSPK Master enter Aux-in 48K PCM Encoder mode |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | SPK is slave role |

Description:

This action is used for NSPK master enter AUX-IN 48K PCM encoder mode.

[[Return to Action Table]](#_Action:)

#### Action 0xE8:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xE8 | NSPK Master exit Aux-in PCM Encoder mode |

Description:

This action is used for NSPK master exit AUX-IN PCM encoder mode.

[[Return to Action Table]](#_Action:)

#### Action 0xE9:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xE9 | NSPK Master enter Aux-in SBC Encoder mode |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | SPK is slave role |

Description:

This action is used for NSPK master enter AUX-IN SBC encoder mode.

[[Return to Action Table]](#_Action:)

#### Action 0xEB:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xEB | NSPK dynamic creation |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | CSB state and Line-in state doesn’t under IDLE mode |

Description:

This action is used for NSPK dynamic creation.

[[Return to Action Table]](#_Action:)

#### Action 0xEC:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xEC | NSPK switch channel |

Description:

This action is used for NSPK switch channel.

[[Return to Action Table]](#_Action:)

#### Action 0xED:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xED | nSPK power off all speakers |

Description:

This action is used to execute the power off process. If nSPK is connected, this action is used to disconnect the remote nSPK first then MCU receive EVENT [BTM\_Utility\_Rsp](#_BTM_Utility_Req_(0x1B)) (0x1B) with action\_type 0x05.

After receive the event, MCU can send MMI\_Action [0x53](#_Action_0x53:), [0x54](#_Action_0x54:) to power off current SPK.

[[Return to Action Table]](#_Action:)

#### Action 0xEE:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xEE | NSPK AFH SBC ENCODING AUDIO SYNC |

Description:

This action is used for NSPK AFH SBC encoding audio sync.

[[Return to Action Table]](#_Action:)

#### Action 0xF4:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xF4 | NSPK ENTER NSPK MODE |

Description:

This action is used for NSPK enter NSPK mode. It reports nSPK link status and CSB state.

[[Return to Action Table]](#_Action:)

#### Action 0xF5:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xF5 | NSPK ENTER BROADCAST MODE |

Description:

This action is used for NSPK enter Broadcast mode. It reports nSPK link status and CSB state.

[[Return to Action Table]](#_Action:)

#### Action 0xF6:

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xF6 | NSPK ADD THIRD SPK |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | SPK is slave role |

Description:

This action is used to add third SPK.

[[Return to Action Table]](#_Action:)

#### Action 0xF7

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xF7 | NSPK\_SOUND\_SYNCHRONIZATION |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | SPK is slave role |

Description:

This action is used for NSPK sound re-synchronize

[[Return to Action Table]](#_Action:)

#### Action 0xF8

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xF8 | NSPK\_CSB\_CONNECTED\_MODE\_SWITCH |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | SPK is slave role |

Description:

This action is used for NSPK to switch connected mode.

[[Return to Action Table]](#_Action:)

#### Action 0xF9

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xF9 | NSPK\_BACK\_TO\_LAST\_MODE |

Return error: Length: 1 Byte

|  |  |  |
| --- | --- | --- |
| Value | Description | Condition |
| 0x01 | Command disallow | CSB state doesn’t under IDLE mode |

Description:

This action is used for NSPK to switch back to last mode.

[[Return to Action Table]](#_Action:)

# Events

## Command\_Ack (0x00)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Command\_Ack | 0x00 | Command\_ID, Status |

**Description:**

This event is used to acknowledge the command. Status field of this event will tell whether the command is processed successfully or not.

**Event Parameters:**

Command\_ID: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | The Command ID of the command to acknowledge |

Status: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Command complete: BTM can handle this command. |
| 0x01 | Command disallow: BTM cannot handle this command. |
| 0x02 | Unknown command |
| 0x03 | Parameters error |
| 0x04 | BTM is busy:  This status is used to notify host MCU that SPP data cannot be sent out in this moment because of ACL Tx buffer or RFCOMM credit issue. BTM will send Command\_Ack event with the status "Command complete" once the SPP data can be processed. |
| 0x05 | BTM memory is full:  This status is used to notify host MCU that SPP/BLE data cannot be sent out in this moment because of OS heap memory is full. BTM will send Command\_Ack event with the status "Command complete" once the SPP data can be processed and MCU must resend previous packet. |
| Others | Reserved |

[[Return to event Table]](#_UART_Event)

## BTM\_Status (0x01)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| BTM\_Status | 0x01 | State, Link\_Info, Parameter |

**Description:**

This event is used to indicate the BTM status. When there is any change in the BTM status by using this event status change will be informed to the host MCU.

**Event Parameters:**

State: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Power OFF state |
| 0x01 | Pairing state (discoverable mode) |
| 0x02 | Power ON state |
| 0x03 | Pairing successful |
| 0x04 | Pairing failed |
| 0x05 | HF/HS link established |
| 0x06 | A2DP link established |
| 0x07 | HF link disconnected |
| 0x08 | A2DP link disconnected |
| 0x09 | SCO link connected |
| 0x0A | SCO link disconnected |
| 0x0B | AVRCP link established |
| 0x0C | AVRCP link disconnected |
| 0x0D | Standard SPP connected |
| 0x0E | Standard\_SPP / iAP disconnected |
| 0x0F | Standby state |
| 0x10 | iAP connected |
| 0x11 | ACL disconnected |
| 0x12 | MAP connected |
| 0x13 | MAP operation forbidden |
| 0x14 | MAP disconnected |
| 0x15 | ACL connected |

Link\_Info: For State 0x00, 0x01, 0x0F, 0x12, 0x13, 0x14 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reserve |

Link\_Info: For State 0x02 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | 0: Power on  1: Already power on. |

Link\_Info: For State 0x03, 0x09, 0x0A Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Current link id |

Link\_Info: For State 0x04 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | For pairing failure(0x04) case, This parameter indicates the reason  0: Time out  1: Fail |

Link\_Info: For State 0x05 0x06 0x0B Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | This parameter indicates both linked device and data base information.  The format is shown below:  Bit7~4 : linked device id(0~7)  Bit3~0 : linked data base(0 or 1) |

Link\_Info: For State 0x07 0x08 0x0C 0x15 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | This parameter shows the linked data base (0 or 1) |

Link\_Info: For State 0x0D 0x10 Length: 5~19 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Byte 0 This parameter indicates both link device and data base information. The format is shown below: Bit7~4 : linked device id(0~7) Bit3~0 : linked data base(0, 1, 2) |
| 0xXX | Byte 1: channel index Specify this byte for data transmission in 2 SPP or 2 iAP connection condition. MCU can just copy this byte to channel\_index byte of Send\_SPP/iAP\_Data if MCU want to transfer data to remote APP Bit0~1: Bluetooth connection index (data base index). Range from 0~3. Bit 2: LE connection indicator Bit3~5: rfcomm index. Range from 0 ~ 7. Bit6~7: iAP session index. 1 is 1st session and 2 is 2nd session. 0 means not a iAP / iAP2 connection |
| 0xXX | Byte 2: UUID length. 2 or 16 bytes UUID |
| 0xXXXXXXXXXX | UUID |

Link\_Info: For State 0x0E Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Byte 1: channel index Specify this byte for data transmission in 2 SPP or 2 iAP connection condition. MCU can just copy this byte to channel\_index byte of Send\_SPP/iAP\_Data if MCU want to transfer data to remote APP Bit0~1: Bluetooth connection index (data base index). Range from 0~3. Bit 2: LE connection indicator Bit3~5: rfcomm index. Range from 0 ~ 7. Bit6~7: iAP session index. 1 is 1st session and 2 is 2nd session. 0 means not a iAP / iAP2 connection |

Link\_Info: For State 0x11 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Disconnection |
| 0x01 | Link loss |

Parameter: For State 0x15 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | HSP is connected |
| 0x01 | HFP is connected |

[[Return to event Table]](#_UART_Event)

## Call\_Status (0x02)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Call\_Status | 0x02 | Data\_Base\_Index, Call\_Status |

**Description:**

This event is used to indicate about the HF call status of BTM.

**Event Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for dedicate link |
| 0x01 | database 1 for dedicate link |

Call\_Status: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Idle |
| 0x01 | voice dial |
| 0x02 | incoming call |
| 0x03 | outgoing call |
| 0x04 | call active |
| 0x05 | a call active with a call waiting |
| 0x06 | a call active with a call hold |

[[Return to event Table]](#_UART_Event)

## Caller\_ID (0x03)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Caller\_ID | 0x03 | Data\_Base\_Index, Number |

**Description:**

This event is used to indicate about the caller ID of the incoming call.

**Event Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for dedicate link |
| 0x01 | database 1 for dedicate link |

Number: Length: N Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX.. | caller Id or phone number |

[[Return to event Table]](#_UART_Event)

## SMS\_Received\_Indication (0x04)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| SMS\_Received\_Indication | 0x04 | Data\_Base\_Index, Indication |

**Description:**

This event is used to indicate about the sms status that BTM received from mobile phone.

**Event Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for dedicate link |
| 0x01 | database 1 for dedicate link |

Indication: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | No new sms received |
| 0x01 | New sms received |

[[Return to event Table]](#_UART_Event)

## Missed\_Call\_Indication (0x05)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Missed\_Call\_Indication | 0x05 | Data\_Base\_Index, Information |

**Description:**

This event is used to indicate that BTM received missed call notification from mobile phone.

**Event Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for dedicate link |
| 0x01 | database 1 for dedicate link |

Information: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reserved |

[[Return to event Table]](#_UART_Event)

## Phone\_Max\_Battery\_Level (0x06)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Phone\_Max\_Battery\_Level | 0x06 | Data\_Base\_Index, Max\_Battery\_Level |

**Description:**

This event is used to indicate about the mobile phone max battery level that BTM received from mobile phone.

**Event Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for dedicate link |
| 0x01 | database 1 for dedicate link |

Max\_Battery\_Level: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Max Battery Level |

[[Return to event Table]](#_UART_Event)

## Phone\_Current\_Battery\_Level (0x07)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Phone\_Current\_Battery\_Level | 0x07 | Data\_Base\_Index, Battery\_Level |

**Description:**

This event is used to indicate about the mobile phone current battery level that BTM received from mobile phone.

**Event Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for dedicate link |
| 0x01 | database 1 for dedicate link |

Battery\_Level: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Current Battery Level |

[[Return to event Table]](#_UART_Event)

## Roaming\_Status (0x08)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Roaming\_Status | 0x08 | Data\_Base\_Index, Status |

**Description:**

This event is used to indicate about the roaming status that BTM received from mobile phone.

**Event Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for dedicate link |
| 0x01 | database 1 for dedicate link |

Status: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Non Roaming |
| 0x01 | Roaming |

[[Return to event Table]](#_UART_Event)

## Phone\_Max\_Signal\_Strength\_Level (0x09)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Phone\_Max\_Signal\_Strength\_Level | 0x09 | Data\_Base\_Index, Signal\_Level |

**Description:**

This event is used to indicate the max signal strength level that BTM received from mobile phone.

**Event Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for dedicate link |
| 0x01 | database 1 for dedicate link |

Signal\_Level: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Phone’s maximum signal strength |

[[Return to event Table]](#_UART_Event)

## Phone\_Current\_Signal\_Strength\_Level (0x0A)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Phone\_Current\_Signal\_Strength\_Level | 0x0A | Data\_Base\_Index, Signal\_Level |

**Description:**

This event is used to indicate the current signal strength level that BTM received from mobile phone.

**Event Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for dedicate link |
| 0x01 | database 1 for dedicate link |

Signal\_Level: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Phone’s current signal strength |

[[Return to event Table]](#_UART_Event)

## Phone\_Service\_Status (0x0B)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Phone\_Service\_Status | 0x0B | Data\_Base\_Index, Service |

**Description:**

This event is used to indicate about the service status that BTM received from mobile phone.

**Event Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for dedicate link |
| 0x01 | database 1 for dedicate link |

Service: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | No Service Available |
| 0x01 | Service Available |

[[Return to event Table]](#_UART_Event)

## BTM\_Battery\_Status (0x0C)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| BTM\_Battery\_Status | 0x0C | Battery\_Status, Voltage\_Level |

**Description:**

This event is used to indicate about the BTM's battery status.

**Event Parameters:**

Battery\_Status: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | dangerous level, and will auto shutdown |
| 0x01 | low level |
| 0x02 | normal level |
| 0x03 | high level |
| 0x04 | full level |
| 0x05 | in charging |
| 0x06 | charging completed |

Voltage\_Level: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | batt < 3.1V |
| 0x01 | 3.1V <= batt < 3.2V |
| 0x02 | 3.2V <= batt < 3.3V |
| 0x03 | 3.3V <= batt < 3.4V |
| 0x04 | 3.4V <= batt < 3.5V |
| 0x05 | 3.5V <= batt < 3.6V |
| 0x06 | 3.6V <= batt < 3.7V |
| 0x07 | 3.7V <= batt < 3.8V |
| 0x08 | 3.8V <= batt < 3.9V |
| 0X09 | 3.9V <= batt < 4.0V |
| 0X0A | 4.0V <= batt < 4.1V |
| 0X0B | 4.1V <= batt < 4.2V |
| 0x0C | 4.2V <= batt |

[[Return to event Table]](#_UART_Event)

## BTM\_Charging\_Status (0x0D)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| BTM\_Charging\_Status | 0x0D | Charger\_Status, Charger\_Type |

**Description:**

This event is used to indicate about the charger status and charger type.

**Event Parameters:**

Charger\_Status: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Charger is not plugged in |
| 0x01 | In charging |
| 0x02 | Charging completed |
| 0x03 | Charging failed |
| 0x04 | Charger type report |

Charger\_Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | BC\_TYPE\_UNKNOWN |
| 0x01 | BC\_TYPE\_NON\_DCD |
| 0x02 | BC\_TYPE\_SDP |
| 0x03 | BC\_TYPE\_DCP |
| 0x04 | BC\_TYPE\_CDP |
| 0x10 | BC\_TYPE\_SONY |
| 0x20 | BC\_TYPE\_APPLE\_2\_5W |
| 0x21 | BC\_TYPE\_APPLE\_5W |
| 0x22 | BC\_TYPE\_APPLE\_10W |
| 0x23 | BC\_TYPE\_APPLE\_12W |
| Others | Reserved |

[[Return to event Table]](#_UART_Event)

## Reset\_To\_Default (0x0E)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Reset\_To\_Default | 0x0E | Reserved |

**Description:**

This event is used to indicate that BTM finished the Master Reset for the MMI command (0x56) trigger.

**Event Parameters:**

Reserved: Length: 2 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX | database 0 for dedicate link |

[[Return to event Table]](#_UART_Event)

## Report\_HF\_Gain\_Level (0x0F)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_HF\_Gain\_Level | 0x0F | Data\_Base\_Index, Gain\_Level |

**Description:**

This event is used to report the HF gain level set by remote Audio Gateway (Phone).

**Event Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for a dedicate HF/HS device |
| 0x01 | database 1 for a dedicate HF/HS device |

Gain\_Level: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00-0x0F | Gain level that synched with HF device |

[[Return to event Table]](#_UART_Event)

## EQ\_Mode\_Indication (0x10)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| EQ\_Mode\_Indication | 0x10 | EQ\_Mode,Reserved |

**Description:**

This event is used to notify the host MCU that EQ mode setting changed by MMI or EQ\_Mode\_Setting command.

**Event Parameters:**

EQ\_Mode Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Off Mode |
| 0x01 | Soft Mode |
| 0x02 | Bass Mode |
| 0x03 | Treble Mode |
| 0x04 | Classical Mode |
| 0x05 | Rock Mode |
| 0x06 | Jazz Mode |
| 0x07 | Pop Mode |
| 0x08 | Dance Mode |
| 0x09 | R&B Mode |
| 0x0B | User Mode 1 |

Reserved: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reserved |

[[Return to event Table]](#_UART_Event)

## Read\_Linked\_Device\_Information\_Reply (0x17)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Read\_Linked\_Device\_Information\_Reply | 0x17 | Data\_Base\_Index, Type, Info |

**Description:**

This event is used to reply Read\_Linked\_Device\_Information command.

**Event Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for dedicate link |
| 0x01 | database 1 for dedicate link |

Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | reply device name |
| 0x01 | reply in-band ringtone status |
| 0x02 | reply remote device is a iAP device or standard SPP device |
| 0x03 | reply remote device supported AVRCP function |
| 0x04 | reply HF&A2DP gain |
| 0x05 | reply Line In gain |
| 0x06 | reply A2DP used codec type |
| others | reserved |

Info: for Type 0x00 Length: N Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX… | N bytes Bluetooth name with NULL terminated. (N <= 249 with NULL terminated) Note: If remote device response empty name string, then BTM will report name with data NULL terminated (0x00) only. |

Info: for Type 0x01 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Disabled |
| 0x01 | Enabled |

Info: for Type 0x02 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Standard SPP device |
| 0x01 | iAP device |

Info: for Type 0x03 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Bit Mask  bit0:media player status notification bit1:absolute volume control |

Info: for Type 0x04 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | bit[3:0]: A2DP gain bit[7:4]: HF gain |

Info: for Type 0x05 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Line in gain |

Info: for Type 0x06 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | SBC |
| 0x02 | AAC |
| 0xFF | Vendor (LDAC) |
| Others | Reserved |

[[Return to event Table]](#_UART_Event)

## Read\_BTM\_Version\_Reply (0x18)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Read\_BTM\_Version\_Reply | 0x18 | Type, Version |

**Description:**

This event is used to reply Read\_BTM\_Version command.

**Event Parameters:**

Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | UART Version |
| 0x01 | BTM FW Version |

Version: for Type 0x00 Length: 2 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX | 1st byte: UART Command Main version  2nd byte: UART Command Sub version  for example 00 07 means version 0.07 |

Version: for Type 0x01 Length: 2 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX | 1st byte: Firmware Main version  2nd byte: Firmware Sub version  for example 00 07 means version 0.07 |

[[Return to event Table]](#_UART_Event)

## Call\_List\_Report (0x19)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Call\_List\_Report | 0x19 | Call\_List\_String (ASCII) |

**Description:**

This event is used to notify +CLCC information for the MMI command (0x0F).

**Event Parameters:**

Call\_List\_String: Length: N Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXXXX… | “+CLCC: xxxxx” |

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Access\_Finish | 0x16 | Type, Status |

Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Report call list finish |

Status: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Successful |
| Others | Error: when AG do not support this command, it will response +ERROR, and then SPK Module will report this to MCU |

[[Return to event Table]](#_UART_Event)

## AVC\_Vendor\_Dependent\_Response (0x1A)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| AVC\_Vendor\_Dependent\_Response | 0x1A | Data\_Base\_Index, AVC\_Rsp\_Payload |

**Description:**

This event is used to reply AVC\_Vendor\_Dependent\_Cmd command. Detail response information needs to refer to AVRCP Specification [[4]](#_References)

**Event Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for dedicate link |
| 0x01 | database 1 for dedicate link |

AVC\_Rsp\_Payload: Length: N Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| byte0 | Response type:  0x08: Not implement  0x09: Accept  0x0A: Reject  0x0C: Stable  0x0D: Changed  0x0F: Interim |
| Byte1 | Subunit\_type:5 bit | Subunit\_ID:3 bit: 0x48 |
| Byte2 | Opcode: 0x00 (Vendor Dependent) |
| Byte3~5 | Company ID: 0x00 0x19 0x58 |
| Byte6 | AVRCP1.3 AVC specific command PDU Id |
| Byte7 | Packet Type: 0x00 (single packet) |
| Byte8~9 | Response parameter length |
| Byte10-N | Response parameter |

[[Return to event Table]](#_UART_Event)

## BTM\_Utility\_Req (0x1B)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| BTM\_Utility\_Req | 0x1B | Action\_Type, Parameter |

**Description:**

This event is used to request host MCU to do some specific function.

**Event Parameters:**

Action\_Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | BTM ask MCU to control the external amplifier |
| 0x01 | BTM report the Aux line-in status to Host MCU. |
| 0x02 | BTM notify MCU to handle BTM or MCU update process |
| 0x03 | BTM notify MCU eeprom update finish |
| 0x04 | BTM report the A2DP codec status to Host MCU. |
| 0x05 | [NSPK] BTM notify MCU to sync power off BTM |
| 0x06 | [NSPK] BTM notify MCU to sync Volume Control |
| 0x07 | [NSPK] BTM notify MCU to sync internal gain |
| 0x08 | [NSPK] BTM notify MCU to sync A2DP absolute volume |
| 0x09 | [NSPK] BTM notify MCU current channel setting |
| 0x0A | [NSPK] BTM notify MCU synced NSPK power condition |
| 0x0B | [NSPK] BTM notify MCU NSPK command success |
| 0x0C | [NSPK] BTM notify MCU NSPK command fail |
| 0x0D | [NSPK] BTM notify MCU certain NSPK Slave status has been changed |
| 0x0E | Reserved |
| 0x0F | Reserved |
| 0x10 | Reserved |
| 0x11 | [NSPK] BTM notify MCU to sync Line-in absolute volume |
| 0x12 | [NSPK] BTM notify MCU that NSPK connection complete. |
| others | reserved |

Parameter: For Action\_Type 0x00 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Mute or switch off amplifier |
| 0x01 | Unmute or switch on amplifier |
| Others | Reserved |

Parameter: For Action\_Type 0x01 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Aux line in is unplugged. |
| 0x01 | Aux line in is plugged. |
| 0x02 | Aux line in is plugged and with audio signal. |
| 0x03 | Aux line in is plugged and silence. |
| Others | Reserved |

Parameter: For Action\_Type 0x02 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | BTM FW update |
| 0x01 | MCU FW update |
| Others | Reserved |

Parameter: For Action\_Type 0x03 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | eeprom update successful |
| Others | Reserved |

Parameter: For Action\_Type 0x04 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | A2DP stop |
| 0x01 | A2DP start |
| Others | Reserved |

Parameter: For Action\_Type 0x06 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Volume Up |
| 0x01 | Volume Down |
| Others | Reserved |

Parameter: For Action\_Type 0x07 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x0X | Fist 4 bits indicate A2DP gain level |
| 0xX0 | Last 4 bits indicate Line In gain level |

Parameter: For Action\_Type 0x08 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | A2DP Absolute Volume |

Parameter: For Action\_Type 0x09 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | No Mix channel |
| 0x01 | L+L channel |
| 0x02 | R+R channel |
| 0x03 | (L+R) / 2 -> L, (L+R) / 2 -> R channel |

Parameter: For Action\_Type 0x0A Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Battery power with Low battery |
| 0x01 | Adaptor power with Low battery |
| 0x02 | Battery power |
| 0x03 | Adaptor power |

Parameter: For Action\_Type 0x0B 0x0C Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Audio effect type |

Parameter: For Action\_Type 0x0D Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reserved |

Parameter: For Action\_Type 0x11 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Line-in Absolute Volume |

Parameter: For Action\_Type 0x12 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x01 | NSPK connect complete |

[[Return to event Table]](#_UART_Event)

## Vendor\_AT\_Cmd\_Rsp (0x1C)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Vendor\_AT\_Cmd\_Rsp | 0x1C | Data\_Base\_Index, Status |

**Description:**

This event is used to reply Vendor\_AT\_Cmd (0x0A) command.

**Event Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for dedicate link |
| 0x01 | database 1 for dedicate link |

Status: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | AG response OK |
| 0x01 | AG response ERROR |
| 0x02 | No response from AG |
| Others | Reserved |

[[Return to event Table]](#_UART_Event)

## Report\_Vendor\_AT\_Event (0x1D)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_Vendor\_AT\_Event | 0x1D | Data\_Base\_Index, Result\_Payload |

**Description:**

This event is used to reply Vendor\_AT\_Cmd (0x0A) command.

**Event Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for dedicate link |
| 0x01 | database 1 for dedicate link |

Result\_Payload: Length: N Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX… | Result code. For example : AG send result code as "+test:1" , the result code will be "+test:1" |

[[Return to event Table]](#_UART_Event)

## Read\_Link\_Status\_Reply (0x1E)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Read\_Link\_Status\_Reply | 0x1E | Device\_State, Database0\_Connect\_Status, Database1\_Connect\_Status, Database0\_Play\_Status Database1\_Play\_Status, Database0\_Stream\_Status, Database1\_Stream\_Status, |

**Description:**

This event is used to reply the Read\_Link\_Status(0X0D) command.

**Event Parameters:**

Device\_State: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Power OFF state |
| 0x01 | pairing state (discoverable mode) |
| 0x02 | standby state |
| 0x03 | Connected state with only HF profile connected |
| 0x04 | Connected state with only A2DP profile connected |
| 0x05 | Connected state with only SPP profile connected |
| 0x06 | Connected state with multi-profile connected |

Database0\_Connect\_Status / Database1\_Connect\_Status: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | 1 indicates connected Bit0 : A2DP profile signaling channel connected Bit1 : A2DP profile stream channel connected Bit2 : AVRCP profile connected Bit3 : HF profile connected Bit4 : SPP connected |

Database0\_Play\_Status / Database1\_Play\_Status: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | STOP |
| 0x01 | PLAYING |
| 0x02 | PAUSED |
| 0x03 | FWD\_SEEK |
| 0x04 | REV\_SEEK |
| 0x05 | FAST\_FWD |
| 0x06 | REWIND |
| 0x07 | WAIT\_TO\_PLAY |
| 0x08 | WAIT\_TO\_PAUSE |

Database0\_Stream\_Status / Database1\_Stream\_Status: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | No Streaming |
| 0x01 | Streaming is going on |

[[Return to event Table]](#_UART_Event)

## Read\_Paired\_Device\_Record\_Reply (0x1F)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Read\_Paired\_Device\_Record\_Reply | 0x1F | Paired\_Device\_Number, Paired\_Record |

**Description:**

This event is used to reply the Read\_Paired\_Device\_Record (0X0E) command.

**Event Parameters:**

Paired\_Device\_Number: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Paired Device Number |

Paired\_Record: (7 Bytes Per Record) Length: (7\*Total Record) Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXXXXXXXXXXXX | Byte 0: Link priority : 1 is the highest(newest device) and 4 is the lowest(oldest device)  Byte 1~6 :Linked device BD address (6 bytes with low byte first) |

[[Return to event Table]](#_UART_Event)

## Read\_Local\_BD\_Address\_Reply (0x20)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Read\_Local\_BD\_Address\_Reply | 0x20 | BD\_Address |

**Description:**

This event is used to reply the Read\_Local\_BD\_Address (0X0F) command.

**Event Parameters:**

BD\_Address: Length: 6 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX.. | BD address with lower byte first |

[[Return to event Table]](#_UART_Event)

## Read\_Local\_Device\_Name\_Reply (0x21)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Read\_Local\_Device\_Name\_Reply | 0x21 | Name\_Length, Device\_Name |

**Description:**

This event is used to reply the Read\_Local\_Device\_Name (0X10) command.

**Event Parameters:**

Name\_Length: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX.. | Name Length |

Device\_Name: Length: Name\_Length Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX.. | Device Name With Length Name Length |

[[Return to event Table]](#_UART_Event)

## Reprt\_SPP/iAP/LE\_Data (0x22)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_SPP/iAP/LE\_Data | 0x22 | Channel\_Index, Type,  Total\_Length, Payload\_Length, Payload |

**Description:**

This event is used to send SPP/iAP /LE data coming from remote device to host MCU.

**Event Parameters:**

Channel\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x01 | Specify this byte for data transmission in 2 SPP / 2 iAP / LE connection. bit0~1: Bluetooth connection index (data base index). Range from 0~3. bit2: LE connection indicator bit3~5: rfcomm index. Range from 0 ~ 7. bit6~7: iAP session index. 1 is 1st session and 2 is 2nd session. 0 means not a iAP / iAP2 connection |

Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | single packet |
| 0x01 | fragmented start packet |
| 0x02 | fragmented continue packet |
| 0x03 | fragmented end packet |

Total\_Length: Length: 2 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX | total payload length |

Payload\_Length: Length: 2 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX | payload length in this packet |

Payload: Length: N Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX… | payload |

[[Return to event Table]](#_UART_Event)

## Reprt\_Link\_Back\_Status (0x23)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_Link\_Back\_Status | 0x23 | Link\_Back\_Status, Link\_Back\_Result |

**Description:**

This event is used to indicate the link back status of BTM..

**Event Parameters:**

Link\_Back\_Status: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | ACL connection |
| 0x01 | HF connection |
| 0x02 | A2DP connection |
| 0x03 | SPP connection |
| ~~0x04~~ | ~~under page state~~ |

Link\_Back\_Result: for Link\_Back\_Status 0x00 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | 0xFF : fail Others : success Bit7~4 : linked device id(0~7) Bit3~0 : linked data base(0 or 1) |

Link\_Back\_Result: for Link\_Back\_Status 0x01~0x03 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Success |
| 0x01 | Failed |

[[Return to event Table]](#_UART_Event)

## Ringtone\_Status\_Indication (0x24)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Ringtone\_Status\_Indication | 0x24 | Ringtone\_Mode, Status |

**Description:**

This event is used to indicate the ringtone mode and ringtone playback status.

**Event Parameters:**

Ringtone\_Mode: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Please Refer [Tone Type](#_BTM_Utility_Function_(0x13)) |

Status: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Ringtone playback is going to be stopped |
| 0x01 | Ringtone playback is going to start |

[[Return to event Table]](#_UART_Event)

## User\_Confirm\_SSP\_Req (0x25)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| User\_Confirm\_SSP\_Req | 0x25 | Data\_Base\_Index, Numeric\_Value |

**Description:**

This event is used to indicate that user confirmation of a numeric value is required. The host shall reply with the User\_Confirm\_SSP\_Req\_Reply command to indicate whether user has confirmed the numeric value or not. If the host has output capability it shall display the Numeric\_Value to the user. It shall reply the yes/no response from the user.

**Event Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 for dedicate link |
| 0x01 | database 1 for dedicate link |

Numeric\_Value: Length: 4 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00000000~0x000F423F | Numeric value to be displayed. Valid values are decimal 000000 – 999999. |

[[Return to event Table]](#_UART_Event)

## Report\_AVRCP\_Volume\_Ctrl (0x26)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_AVRCP\_Volume\_Ctrl | 0x26 | Data\_Base\_Index, Volume\_Ctrl\_Indication |

**Description:**

This event is used to indicate the received AVRCP volume control to MCU. For AVRCPv1.0, remote device may request speaker to adjust volume up or down.

**Event Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 that related to a dedicate A2DP link |
| 0x01 | database 1 that related to a dedicate A2DP link |

Volume\_Ctrl\_Indication: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | volume up |
| 0x01 | volume down |

[[Return to event Table]](#_UART_Event)

## Report\_Input\_Signal\_Level (0x27)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_Input\_Signal\_Level | 0x27 | Report\_Mask\_P0, Report\_Mask\_P1, Report\_Mask\_P2, Report\_Mask\_P3, Input\_level\_P0, Input\_level\_P1, Input\_level\_P2, Input\_level\_P3, |

**Description:**

This event shall be send under two condition, first condition is when the MCU set the specific GPIOs as input by GPIO\_CTRL(0x1E) command and the second condition is when input GPIO input signal level is changed.

**Event Parameters:**

Report\_Mask\_P0: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0bXXXXXXXX | Bit mask of P0 for input signal level reporting.  Bit 0: P0\_0 input signal level reporting indication Bit 1: P0\_1 input signal level reporting indication …… |

Report\_Mask\_P1: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0bXXXXXXXX | Bit mask of P1 for input signal level reporting.  Bit 0: P1\_0 input signal level reporting indication Bit 1: P1\_1 input signal level reporting indication …… |

Report\_Mask\_P2: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0bXXXXXXXX | Bit mask of P2 for input signal level reporting.  Bit 0: P2\_0 input signal level reporting indication Bit 1: P2\_1 input signal level reporting indication …… |

Report\_Mask\_P3: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0bXXXXXXXX | Bit mask of P3 for input signal level reporting.  Bit 0: P3\_0 input signal level reporting indication Bit 1: P3\_1 input signal level reporting indication …… |

Input\_level\_P0: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0bXXXXXXXX | Input signal level reporting of P0.  Bit 0: P0\_0 input signal level. Bit 1: P0\_1 input signal level. …… |

Input\_level\_P1: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0bXXXXXXXX | Input signal level reporting of P1.  Bit 0: P1\_0 input signal level. Bit 1: P1\_1 input signal level. …… |

Input\_level\_P2: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0bXXXXXXXX | Input signal level reporting of P2.  Bit 0: P2\_0 input signal level. Bit 1: P2\_1 input signal level. …… |

Input\_level\_P3: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0bXXXXXXXX | Input signal level reporting of P3.  Bit 0: P3\_0 input signal level. Bit 1: P3\_1 input signal level. …… |

[[Return to event Table]](#_UART_Event)

## Report\_iAP\_Info (0x28)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_iAP\_Info | 0x28 | Type, Parameter |

**Description:**

This event is used to report the iAP data session status.

**Event Parameters:**

Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Data session status. |
| 0x01 | Authentication completed. |

Parameter: For Type 0x00 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Bit0~3:data session status  0x00:data session close  0x01:data session open Bit4~7:(only valid for session open)  protocol index |

Parameter: For Type 0x01 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | iAP authentication OK |
| 0x01 | iAP2 authentication OK |

[[Return to event Table]](#_UART_Event)

## Report\_AVRCP\_ABS\_Volume\_Level (0x29)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_AVRCP\_ABS\_Volume\_Level | 0x29 | Data\_Base\_Index, Absolute\_Volume |

**Description:**

This event is used to indicate the received AVRCP absolute volume level to MCU. For AVRCPv1.0, remote device may request speaker to change the volume level.

**Event Parameters:**

Data\_Base\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | database 0 that related to a dedicate A2DP link |
| 0x01 | database 1 that related to a dedicate A2DP link |

Absolute\_Volume: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Report in range 0x00~0x7F to indicate the percentage of total(max) volume level |

[[Return to event Table]](#_UART_Event)

## Report\_Voice\_Prompt\_Status (0x2A)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_Voice\_Prompt\_Status | 0x2A | Status, Reserved |

**Description:**

This event is used to report the TTS status.

**Event Parameters:**

Status: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x01 | Ready |
| Others | Reserved |

Reserved: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reserved |

[[Return to event Table]](#_UART_Event)

## Report\_Type\_Codec (0x2D)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_Type\_Codec | 0x2D | Sampling\_Frequency, Mode |

**Description:**

This event is used to inform MCU about the next I2S state which DSP prepare to enter. So that MCU can configure its external CODEC to corresponding state.

**Event Parameters:**

Samping\_Frequency: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | 8KHz sample rate |
| 0x02 | 16KHz sample rate |
| 0x04 | 32KHz sample rate |
| 0x05 | 48KHz sample rate |
| 0x06 | 44.1KHz sample rate |
| 0x07 | 88KHz sample rate |
| 0x08 | 96KHz sample rate |
| Others | Reserved |

Mode: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | No action |
| 0x01 | Prepare |
| 0x02 | Audio in mode |
| 0x03 | PCM mode |
| 0x04 | A2DP decode mode |
| 0x06 | SCO(HF) mode |
| 0x07 | Tone |
| 0x08 | Voice prompt |
| Others | Reserved |

[[Return to event Table]](#_UART_Event)

## Report\_Type\_BTM\_Settings (0x2E)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_Type\_BTM\_Settings | 0x2E | Setting\_Type, Setting\_Value |

**Description:**

This event is used to report the BTM setting status to MCU.

**Event Parameters:**

Setting\_Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Pairing timeout setting |
| 0x01 | Supported codec type setting |
| 0x02 | BTM standby mode setting |
| Others | Reserved |

Setting\_Value: For Setting\_Type 0x00 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reply pairing timeout setting with unit 30 second |

Setting\_Value: For Setting\_Type 0x01 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reply supported codec type with bit mask: bit0:SBC (Mandatory, must always been set) bit1:AAC bit2:VENDOR |

Setting\_Value: For Setting\_Type 0x02 Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | BTM Standby mode disable |
| 0x01 | BTM Standby mode enable |
| Others | Reserved |

[[Return to event Table]](#_UART_Event)

## Report\_MCU\_Update\_Reply (0x2F)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_MCU\_Update\_Reply | 0x2F | Action, Data |

**Description:**

This event is to reply MCU\_Update\_Cmd(0x26).

**Event Parameters:**

Action: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | reply total update Bin size, data(4bytes) |
| 0x01 | reply start or continue Bin data |
| 0x02 | reply last Bin data |
| others | reserved |

[[Return to event Table]](#_UART_Event)

## Report\_BTM\_Initial\_Status (0x30)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_BTM\_Initial\_Status | 0x30 | Status, Reserved |

**Description:**

This event is to report the initialization status to MCU.

**Event Parameters:**

Action: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | initialization completed |
| Others | Reserved |

Reserved: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Reserved |

[[Return to event Table]](#_UART_Event)

## LE\_ANCS\_Service\_Event (0x31)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| LE\_ANCS\_Service\_Event | 0x31 | SubEvent\_Type, SubEvent\_Payload |

**Description:**

This event is used to report many kinds of notification that are generated on iOS devices by Bluetooth low-energy link.

**Event Parameters:**

SubEvent\_Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | ANCS Search event |
| 0x01 | ANCS GetNotification Attribute event |
| 0x02 | ANCS Notification Source event |
| 0x03 | ANCS Data Source event |
| 0x04-0xFF | Reserved |

SubEvent\_Payload: For SubEvent\_Type 0x00

**Payload Description:** Report the ANCS search status to Notification Client.

Search Status: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | ANCS [Found](file:///C:\Users\X00132\Documents\UART_CommandSet_v183_20160516.xlsx#'E31-1'!A1) |
| 0x01 | [ANCS Not](file:///C:\Users\X00132\Documents\UART_CommandSet_v183_20160516.xlsx#'E31-2'!A1) Found |

SubEvent\_Payload: For SubEvent\_Type 0x01

**Payload Description:** Report the ANCS GetNotification status to Notification Client.

GetNotification Event Status: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | No Error |
| 0x01 | Invalid Handle |
| 0x02 | Read not Permitted |
| 0x03 | Write not Permitted |
| 0x04 | Invalid PDU |
| 0x05 | Insufficient Authentication |
| 0x06 | Request not Supported |
| 0x07 | Invalid Offset |
| 0x08 | Insufficient Authorization |
| 0x09 | Prepare Queue Full |
| 0x0A | Attribute Not Found |
| 0x0B | Attribute Not Long |
| 0x0C | Insufficient Encryption Key Size |
| 0x0D | Invalid Attribute Value Length |
| 0x0E | Unlikely Error |
| 0x0F | Unlikely Error |
| 0x10 | Unsupported Group Type |
| 0x11 | Insufficient Encryption |
| 0x12-0x7F | Reserved |
| 0x80-0x9F | Reserved for Application Error |
| 0xA0 | ANCS Error Code : Unknown Command (ANCS Spec) |
| 0xA1 | ANCS Error Code : Invalid Command (ANCS Spec) |
| 0xA2 | ANCS Error Code : Invalid Parameters (ANCS Spec) |
| 0xA3 | ANCS Error Code : Action Failed (ANCS Spec) |
| 0xA4-0xFF | Reserved for Application Error |

SubEvent\_Payload: For SubEvent\_Type 0x02

**Payload Description:** Report the data of ANCS notification source to Notification Client.

Notification Source Value: Length: N Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| …… | [Please refer to the Apple Notification Center Service v1.1 document,  the format of a Notification source.](https://developer.apple.com/library/ios/documentation/CoreBluetooth/Reference/AppleNotificationCenterServiceSpecification/AppleNotificationCenterServiceSpecification.pdf) |

SubEvent\_Payload: For SubEvent\_Type 0x03

**Payload Description:** Report the data of ANCS data source to Notification Client.

Data Source Value: Length: N Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| …… | [Please refer to the Apple Notification Center Service v1.1 document,  the format of a Data source.](https://developer.apple.com/library/ios/documentation/CoreBluetooth/Reference/AppleNotificationCenterServiceSpecification/AppleNotificationCenterServiceSpecification.pdf) |

[[Return to event Table]](#_UART_Event)

## LE\_Signaling\_Event (0x32)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| LE\_Signaling\_Event | 0x32 | SubEvent\_Type, SubEvent\_Payload |

**Description:**

This event is used to send Bluetooth low energy signaling event report/response.

**Event Parameters:**

SubEvent\_Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | LE Status Report |
| 0x01 | LE Advertising Control Report |
| 0x02 | LE Connection Parameter Report |
| 0x03 | LE Connection Parameter Update RSP |
| 0x04-0xFF | reserved |

SubEvent\_Payload: For SubEvent\_Type 0x00

**Payload Description:** Report LE status.

**Payload Format:** Connection status, GATT service status

Connection Status: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Standby. |
| 0x01 | Advertising |
| 0x02 | Scanning. |
| 0x03 | Connected. |

GATT service status: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| Bit 0 | 0: Transparent Data Transfer Service Inactive  1: Transparent Data Transfer Service Active |
| Bit 1 | 0: ANCS Service Inactive  1: ANCS Service Active |
| Bit 2~7 | Reserved |

SubEvent\_Payload: For SubEvent\_Type 0x01

**Payload Description:** Report LE Advertising Control status.

Command Status: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Command Succeeded |
| 0x01-0x3F | [Command Failed. (Please refer to BLUETOOTH SPECIFICATION Version 4.0 [Vol 2] Part D, Error Codes on page 339 for a list of error codes and descriptions).](https://www.bluetooth.org/docman/handlers/downloaddoc.ashx?doc_id=229737) |

SubEvent\_Payload: For SubEvent\_Type 0x02

**Payload Description:** Report Current LE Connection Parameters which is set by LE Master Device.

**Payload Format:** Connection Interval, Connection Latency, Supervision Timeout

Connection Interval: Length: 2 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX | Range: 0x0006 to 0x0C80 (7.5ms ~ 4s), value for the connection event interval. |

Connection Latency: Length: 2 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX | Range: 0x0006 to 0x0C80 (7.5ms ~ 4s), value for this connection. |

Supervision Timeout: Length: 2 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX | Range: 0x000A to 0x0C80 (100ms ~ 32s), Supervision timeout for the LE Link |

SubEvent\_Payload: For SubEvent\_Type 0x03

**Payload Description:** Report the LE connection parameter update response.

Result: Length: 2 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x0000 | Connection Parameters Accepted. |
| 0x0001 | Connection Parameters Rejected. |
| Other | Reserved |

[[Return to event Table]](#_UART_Event)

## Report\_nSPK\_Link\_Status (0x33)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_nSPK\_Link\_Status | 0x33 | CSB\_Connection\_State, (Mandatory) CSB\_State, (Mandatory) CSB\_Group\_Number, (Optional) CSB\_Address, (Optional) |

**Description:**

This event is used to indicate the state of Enhanced Connectionless Broadcasting used for NSPK link (eCSB –link).



**SPK3**

**SPK2**

**SPK1**

**Event Parameters:**

CSB\_Connection\_State: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | The master or slave SPK report eCSB-link is non-exist |
| 0x01 | The master SPK1 report SPK1 connect with SPK2 |
| 0x02 | The master SPK1 report SPK1 connect with SPK3 |
| 0x03 | The master SPK1 report SPK1 connect with SPK2 and SPK3 |
| 0x04 | The slave SPK report SPK2 or SPK3 was connected with SPK1 |
| 0x05 | BTM is Broadcast Master and connect to at least one Broadcast Slave |
| 0x06 | BTM is Broadcast Slave and connect to Broadcast Master |
| 0x07 | The master SPK report Timeout while add more slave SPK in Broadcast mode. |

CSB\_State: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | eCSB is standby(paging and page\_scan are disable) |
| 0x01 | eCSB is busy, the eCSB\_paging is be postponed.(BT paging or SCO) |
| 0x02 | eCSB is connecting(the device is in eCSB\_page\_scan or eCSB\_paging mode) |
| 0x03 | Creating new eCSB-link is successful. |
| 0x04 | eCSB-link is loss |
| 0x05 | Power on back to NSPK Master |
| 0x06 | Power on back to NSPK Slave |
| 0x07 | eCSB change from Master connecting to Slave connecting |
| 0x08 | eCSB disconnect by NFC |
| 0x09 | eCSB has connected to SPK1 and connecting to SPK2 |

CSB\_Group\_Number: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | The number of connected slave speaker. This parameter is only valid when CSB\_State is 0x03. |

CSB\_address: Length: 6 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXXXXXXXXXX | This parameter indicates the BT Address of connected CSB device.  Little endian. This parameter is only valid when CSB\_State is 0x03. e.g.  0x102030405060 is 60:50:40:30:20:10 |

[[Return to event Table]](#_UART_Event)

## Report\_nSPK\_Vendor\_Event (0x34)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_nSPK\_Vendor\_Event | 0x34 | SPK\_index, Payload\_length, Payload |

**Description:**

This event is used to report the vendor\_event from remote speaker to MCU



**SPK2**

**SPK1**

**Event Parameters:**

SPK\_Index: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | the slave SPK(SPK2/3) receive vendor cmd form the master SPK(SPK1) |
| 0xFF | When payload length >9 bytes, BTM always returns 0xFF to MCU |
| others | the master SPK(SPK1) receive vendor cmd from the slave SPK |

Payload\_Length: Length: 2 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXX | Payload Length  byte 0:higher byte of length  byte 1:lower byte of length |

*Note: slave speaker only send 1 byte (0x01~0x1F) data to master, the Payload\_length is 0x0001*

Payload: Length: N Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXXXXXXXXX | Vendor command from Master SPK or  Vendor command from Slave SPK (only send 1 byte (0x01~ 0x1F) data) |

[[Return to event Table]](#_UART_Event)

## Report\_nSPK\_Audio\_Setting (0x35)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_nSPK\_Audio\_Setting | 0x35 | Audio\_Effect\_Type, Audio\_Effect\_Value |

**Description:**

This event is used for Slave SPK to report current synced audio setting while Master SPK uses command [0x2C nSPK\_Sync\_Audio\_Effect](#_nSPK_Sync_Audio_Effect_(0x2C)_1).

**Event Parameters:**

Audio\_Effect\_Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Vendor defined audio effect type |

Audio\_Effect\_Value: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Audio effect value |

[[Return to event Table]](#_UART_Event)

## Report\_Sound\_Effect\_Status (0x36)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_Sound\_Effect\_Status | 0x36 | Sound\_Type, Sound\_Status |

**Description:**

This event is used to report sound effect status.

**Event Parameters:**

Sound\_Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | AUDIO EFFECT |
| 0x01 | 3D EFFECT |
| 0x02 | RX NR |
| 0x03 | TX NR |
| others | reserved |

Sound\_Status: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Sound status |

[[Return to event Table]](#_UART_Event)

## Report\_Vendor\_EEPROM\_Data (0x37)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_Vendor\_EEPROM\_Data | 0x37 | Parameter |

**Description:**

This event is used to report Vendor EEPROM data.

**Event Parameters:**

Parameter: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Vendor eeprom data |

[[Return to event Table]](#_UART_Event)

## Report\_IC\_Ver\_Info (0x38)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_IC\_Ver\_Info | 0x38 | Body\_Version, Rom\_Version, Rom\_Sub\_Version, Segment, EEPROM,Table\_Version, EEPROM\_Table\_Sub\_Version, DSP\_Version |

**Description:**

This event is used to report IC Version information.

**Event Parameters:**

Body\_Version: Length:14 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | body version |

Rom\_Version: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Rom Version |

Rom\_Sub\_Version: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Rom Sub-Version |

Segment: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Segment |

EEPROM\_Table\_Version: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | EEPROM Table Version |

EEPROM\_Table\_Sub\_Version: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | EEPROM Table Sub-Version |

DSP\_Version: Length: 2 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | DSP Version |

[[Return to event Table]](#_UART_Event)

## REPORT\_LE\_GATT\_EVENT (0x39)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| REPORT\_LE\_GATT\_EVENT | 0x39 | Sub\_Event\_Type, Parameter |

**Description:**

This event will indicate host about the GATT events from remote device and responses for local Gatt commands. This event contains the sub event for specific gatt events. The sub events are detailed as below:

NOTE: The handle and UUID data is reported in reverse byte order.

### Client\_write\_char\_value (0x00)

This event is used to inform MCU that GATT Client has written a Characteristic Value to **DSPK**.

|  |  |  |
| --- | --- | --- |
| Event | Sub-Event Code | Event Parameters |
| Client\_Write\_Characteristic\_Value | 0x00 | Connection\_Handle, Characteristic\_Value\_Handle, Characteristic\_Value |

**Event Parameters:**

Connection\_Handle: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Connection Handle |

Characteristic\_Value\_Handle: Length: 2 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Characteristic Value Handle |

Characteristic\_Value: Length: N Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Characteristic Value |

### Read\_Local\_Char\_Value\_Res (0x01)

This event is used to report local characteristic value to host. This event is triggered in response to any of the below commands.

* [Read\_Local\_Characteristic\_Value](#_Read_Local_Characteristic_Value_(0x)

|  |  |  |
| --- | --- | --- |
| Event | Sub-Event Code | Event Parameters |
| Read\_Local\_Char\_Value\_Res | 0x01 | Connection\_Handle, Characteristic\_Value\_Handle, Characteristic\_Value |

**Event Parameters:**

Connection\_Handle: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Connection Handle |

Characteristic\_Value\_Handle: Length: 2 Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Characteristic Value Handle |

Characteristic\_Value: Length: N Bytes

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Characteristic Value |

### Discover\_All\_Primary\_Services\_Res (0x02)

This event is used to report a list of all primary services to host. This event is triggered in response to any of the below commands.

* [Read\_Local\_All\_Primary\_Services](#_Read_Local_All_Primary_Services_(0x)

|  |  |  |
| --- | --- | --- |
| Event | Sub-Event Code | Event Parameters |
| Discover\_All\_Primary\_Services\_Res | 0x02 | Connection\_Handle, Attribute\_Data |

**Event Parameters:**

Connection\_Handle: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Connection Handle |

Attribute\_Data: Length: 6 to 20 Bytes

The Attribute Data field is comprised of a list of attribute data.

|  |  |  |  |
| --- | --- | --- | --- |
| Length | Start Group Handle | End Group Handle | Service UUID |
| 1 Byte | 2 Bytes | 2 Bytes | (Length -4) Bytes |

### Discover\_Specific\_Primary\_Service\_Characteristics\_Res (0x03)

This event is used to report a list of all characteristics of a specific service to host. This event is triggered in response to any of the below commands.

* [Read\_Local\_Specific\_Primary\_Service](#_Read_Local_Specific_Primary_Service)

|  |  |  |
| --- | --- | --- |
| Event | Sub-Event Code | Event Parameters |
| Discover\_Specific\_Primary\_Service\_Characteristics\_Res | 0x03 | Connection\_Handle, Length, Attribute\_Data |

**Event Parameters:**

Connection\_Handle: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Connection Handle |

Length: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | The size of each attribute handle-value pair |

Attribute\_Data: Length: 2 to 18 Bytes

The Attribute Data field is comprised of a list of attribute handle and value pairs for characteristic declaration.

|  |  |
| --- | --- |
| Attribute Handle | Attribute Value |
| 2 Bytes | (Length -2) Bytes |

Attribute Value of Characteristic Declaration:

|  |  |  |
| --- | --- | --- |
| Attribute Value | | |
| Characteristic Properties | Characteristic Value Attribute Handle  (2 Bytes) | Characteristic UUID  (2 or 16 Bytes) |

|  |  |
| --- | --- |
| Properties | Value |
| Broadcast | 0x01 |
| Read | 0x02 |
| Write Without Response | 0x04 |
| Write | 0x08 |
| Notify | 0x10 |
| Indicate | 0x20 |
| Authenticated Signed Writes | 0x40 |
| Extended Properties | 0x80 |

### Discover\_All\_Characteristic\_Descriptors\_Res (0x04)

This event is used to report a list of all characteristic descriptors of a specific service to host. This event is triggered in response to any of the below commands.

* [Read\_Local\_Specific\_Primary\_Service](#_Read_Local_Specific_Primary_Service)

|  |  |  |
| --- | --- | --- |
| Event | Sub-Event Code | Event Parameters |
| Discover\_All\_Characteristic\_Descriptors\_Res | 0x04 | Connection\_Handle, Format, Information\_Data |

**Event Parameters:**

Connection\_Handle: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Connection Handle |

Format: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x01 | A list of 1 or more handles with their 16-bit Bluetooth UUIDs |
| 0x02 | A list of 1 or more handles with their 128-bit UUIDs |

Information\_Data: Length: 4 to 20 Bytes

The information data is comprised of a list of data defined in the tables below depending on the value chosen for the format.

|  |  |
| --- | --- |
| Handle | 16-bit Bluetooth UUID |
| 2 Bytes | 2 Bytes |

Format 0x01-handle and 16-bit Bluetooth UUIDs

|  |  |
| --- | --- |
| Handle | 128-bit Bluetooth UUID |
| 2 Octets | 16 Octets |

Format 0x02-handle and 128-bit UUIDs

### Get\_Att\_MTU\_Size\_Res (0x05)

This event is used to report the Att MTU size to host. This event is triggered in response to the below command,

* [Get\_Att\_MTU\_Size](#_Get_Att_MTU_Size_(0x5D))

|  |  |  |
| --- | --- | --- |
| Event | Sub-Event Code | Event Parameters |
| Get\_Att\_MTU\_Size\_Res | 0x05 | Att\_MTU\_Size |

**Event Parameters:**

Att\_MTU\_Size: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | Att MTU Size |

[[Return to event Table]](#_UART_Event)

## Report\_BTM\_Link\_Mode (0x3A)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_BTM\_Link\_Mode | 0x3A | Link\_Mode\_Value, CSB\_Group\_Number |

**Description:**

This event is used to report MCU the last link mode.

**Event Parameters:**

Link\_Mode\_Value: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Single mode |
| 0x01 | Multipoint |
| 0x02 | NSPK master with CSB\_Group\_Number parameter |
| 0x03 | NSPK slave |
| 0x04 | Broadcast Master with CSB\_Group\_Number parameter |
| 0x05 | Broadcast Slave |

CSB\_Group\_Number: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0xXX | The number of connected slave speaker. This parameter is only valid when CSB\_State is 0x03. |

[[Return to event Table]](#_UART_Event)

## DSP\_Dedicated\_Event (0x3B)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| DSP\_Dedicated\_Event | 0x3B | SubEvent Type, SubEvent Payload length, SubEvent Payload |

**Description:**

This event is used to report DSP dedicate data.

**Event Parameters:**

SubEvent Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | SubEvent Type Description |
| 0x00 | Send DSP Metadata to Customer DSP Code. MCU/APP can send data to DSP side with an unfixed data length |
| 0x01 | Customer DSP Code Upload DSP Metadata. MCU/APP can get data from DSP side with an unfixed data length |
| 0x02-0xFF | reserved |

SubEvent Payload Length: Length: 1 Byte

|  |  |
| --- | --- |
| Value | SubEvent Type Description |
| 0~255 | SubEvent Payload Length |

SubEvent Payload: Length: N Bytes

|  |  |
| --- | --- |
| Value | SubEvent Payload Description |
| 0xXX | SubEvent Payload |

[[Return to event Table]](#_UART_Event)

## Report\_nSPK\_MISC\_Event (0x3C)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_nSPK\_MISC\_Event | 0x3C | Type, Parameters |

**Description:**

This event is used to report nSPK MISC Event (can be used by customer to report some special event).

**Event Parameters:**

Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Type Description |
| 0x00 | nspk group status |
| 0x01-0xFF | reserved |

Type = 0

Model Type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameters Description |
| 0 | same model |
| 1 | different model |

Group ID: Length: 4 Bytes

|  |  |
| --- | --- |
| Value | SubEvent Payload Description |
| 0xNNNNNNNN | SHA1 of Master’s BD\_ADDR |

Number of Group: Length: 1 Bytes

|  |  |
| --- | --- |
| Value | SubEvent Payload Description |
| 0-255 | Number of Group |

[[Return to event Table]](#_UART_Event)

## Report\_nSPK\_Exchange\_Link\_info (0x3D)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_nSPK\_Exchange\_Link\_Info | 0x3D | Exchanged\_data |

**Description:**

This event is used to report the exchanged data of CSB link creation. It will be reported after CSB link is connected.

**Event Parameters:**

Exchanged\_data: Length: 16 Bytes

|  |  |
| --- | --- |
| Value | SubEvent Type Description |
| 0xNNNN | 16 bytes exchanged data |

[[Return to event Table]](#_UART_Event)

## Report\_Customized\_Information (0x3E)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_Customized\_Information | 0x3E | payload\_type payload |

**Description:**

This event is used to report the specified information for customization.

**Event Parameters:**

Payload\_type: Length: 1 Byte

|  |  |
| --- | --- |
| Value | SubEvent Type Description |
| 0xNN | used to specified the following payload type |

Payload: Length: N Bytes

|  |  |
| --- | --- |
| Value | SubEvent Payload Description |
| 0xNN | Payload |

[Return to event Table]

## Report\_CSB\_CLK (0x3F)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| Report\_CSB\_CLK | 0x3F | BT Clock |

**Description:**

This event is used to report the BT CLOCK in CSB link. The BT clock is latched if MCU pulls up the CSB\_CLK\_LATCH\_PIN (P2\_7).

**Event Parameters:**

Bt\_clock\_data: Length: 16 Bytes

|  |  |
| --- | --- |
| Value | SubEvent Type Description |
| 0xNNNN | 2 bytes data, high byte is MSB byte |

[Example:](#_UART_Event)

Received raw data “0x3F 0x12 0x34”, the “0x34” is the MSB of BT\_CLOCK, “0x12” is the LSB of BT\_CLOCK

## Report\_Read\_Feature\_List\_Reply (0x40)

|  |  |  |
| --- | --- | --- |
| **Event** | **Event Code** | **Event Parameters** |
| Report\_Read\_Feature\_List\_Reply | 0x40 | Feature0, Feature1, Feature2, Feature3 |

**Description:**

  This event is used to reply the Read\_Feature\_List command to indicate the features supported or not.

**Event Parameters:**

Feature0:                                                                                                                      Length: 1 Byte

|  |  |
| --- | --- |
| **Value** | **Parameter Description** |
| 0xXX | 1 indicates supported; 0 indicates non-supported Bit0 : Stereo Mode Bit1 : Concert Mode Bit2 : Embedded Application Mode Bit3 – Bit7: reserved |

Feature1:                                                                                                                      Length: 1 Byte

|  |  |
| --- | --- |
| **Value** | **Parameter Description** |
| 0xXX | Reserved |

Feature2:                                                                                                                      Length: 1 Byte

|  |  |
| --- | --- |
| **Value** | **Parameter Description** |
| 0xXX | Reserved |

Feature3:                                                                                                                      Length: 1 Byte

|  |  |
| --- | --- |
| **Value** | **Parameter Description** |
| 0xXX | Reserved |

[[Return to event Table]](#_UART_Event)

## REPORT\_TEST\_RESULT\_REPLY (0x41)

|  |  |  |
| --- | --- | --- |
| Event | Event Code | Event Parameters |
| REPORT\_TEST\_RESULT\_REPLY | 0x41 | opcode, status |

**Description:**

This event is used to report the test result of the UART\_CMD\_TEST\_DEVICE.

**Event Parameters:**

opcode: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | CP test result |

status: Length: 1 Byte

|  |  |
| --- | --- |
| Value | Parameter Description |
| 0x00 | Success |
| 0x01 | Error |

[[Return to event Table]](#_UART_Event)

# Revision history

|  |  |  |
| --- | --- | --- |
| Version | Date | History |
| 1.00 | 2016/04/28 | Preliminary Version. Based on UART\_CommandSet\_v177 |
| 1.01 | 2016/08/03 | Base on UART\_CommandSet\_v192 |
| 1.02 | 2016/09/02 | Base on UART\_CommandSet\_v193. |

|  |  |  |
| --- | --- | --- |
| Version | Date | History |
| 2.00 | 2016/10/18 | Preliminary Version. Based on UART\_CommandSet\_v193.  Add error handling response.  Remove similar/no use MMI commands shows in chapter 7.  Remove BTM\_Utility [0x13](#_BTM_Utility_Function_(0x13)_1) I2S function  Add description for UART event packet format. |
| 2.00 | 2017/3/14 | Add [Chapter 2.5.3](#_Error_handling_in) error handling in UART command.  Add new UART command [0x2D](#_LE_GATT_CMD_(0x2D)), [0x2F](#_LE_App_Cmd_(0x2F)), [0x38](#_nSPK_Set_GIAC(0x38)), [0x39](#_READ_FEATURE_LIST_(0x3A)).  Add Line-in Abs\_vol (0x11), connection\_complete (0x12) in event [0x1B](#_BTM_Utility_Req_(0x1B)_1)  Add VP report mode in event [0x2D](#_Report_Type_Codec_(0x2D)_1)  Add CSB\_Connection\_State (0x07) in event [0x33](#_Report_nSPK_Link_Status_(0x33)_1).  Add new UART event [0x40](#_Report_Read_Feature_List_Reply_(0x4).  Modify description for UART command [0x07](#_BTM_Parameter_Setting_(0x07)), [0x13](#_BTM_Utility_Function_(0x13)_1), [0x1E](#_GPIO_CTRL_(0x1E)_1), [0x29](#_LE_Signaling_Cmd_(0x29)_1), [0x30](#_DSP_Runtime_Program__(0x30)_1), [0x35](#_Configure_Vendor_Parameter_(0x35)_1)  Modify subcommand type 0x5 description for UART command [0x17](#_Profile_Link_Back__(0x17)_1).  Modify description for Event [0x01](#_BTM_Status_(0x01)), [0x18](#_Read_BTM_Version_Reply_(0x18)_1), [0x1A](#_AVC_Vendor_Dependent_Response_(0x1A_1), [0x22](#_Reprt_SPP/iAP/LE_Data_(0x22)_1), [0x33](#_Report_nSPK_Link_Status_(0x33)_1)  Remove return error of MMI action ([0x40](#_Action_0x40:))  Remove UART command 0x22, 0x36  Modify [Figure 2.4.1](#_UART_data_exchange)  Update contents table.  Modify nSPK\_Vendor\_Cmd ([0x2A](#_nSPK_Vendor_Cmd_(0x2A)_1)). Extend payload length from **9** to **11** bytes.  Modify Report\_nSPK\_Vendor\_Event ([0x34](#_Report_nSPK_Vendor_Event_(0x34)_1)). BTM returns SPK\_index 0xFF when payload length > 9 bytes |
| 2.00 | 2017/5/4 | Add UART command [0x3A](#_Personal_MSPK_GROUP_Control_(0x3A)) |
| 2.00 | 2017/5/17 | Modify UART command [0x2D](#_LE_GATT_CMD_(0x2D))  Modify UART event [0x39](#_REPORT_LE_GATT_EVENT_(0x39)) |
| 2.00 | 2017/5/23 | Modify UART command [0x12](#_Send_SPP/iAP_Or_LE_Data_(0x12)_1) |
| 2.00 | 2017/5/31 | Modify UART command [0x3A](#_Personal_MSPK_GROUP_Control_(0x3A)) |
| 2.00 | 2017/6/2 | Add new UART event [0x3F](#_Report_CSB_CLK_(0x3F)) |
| 2.00 | 2017/6/9 | Modify UART event [0x3F](#_Report_CSB_CLK_(0x3F)) |
| 2.00 | 2017/6/12 | Modify UART event [0x33](#_Report_nSPK_Link_Status_(0x33)_1),[0x3A](#_Report_BTM_Link_Mode_(0x3A)_1) |
| 2.00 | 2017/6/19 | Add UART command [0x3B](#_UART_CMD_TEST_DEVICE_(0x3B))  Add UART event [0x41](#_REPORT_TEST_RESULT_REPLY_(0x41)) |
| 2.00 | 2017/6/29 | Remove UART event 0x2C  Modify UART command [0x0B](#_AVC_Vendor_Dependent_Cmd_(0x0B))  Modify UART event [0x23](#_Reprt_Link_Back_Status_(0x23)_1) |
| 2.01 | 2017/6/30 | Add UART command 0x40 Event 0x50. Reserve for internal test use. |
| 2.01 | 2017/7/11 | Modify UART command [0x0B](#_AVC_Vendor_Dependent_Cmd_(0x0B)) |
| 2.01 | 2017/7/13 | Modify UART event [0x33](#_Report_nSPK_Link_Status_(0x33)_1),[0x3A](#_Report_BTM_Link_Mode_(0x3A)_1) |

# Remove MMI commands

|  |  |  |
| --- | --- | --- |
| Action | Description | Remark |
| 0x02 | force end active call | Similar with MMI action 0x06 |
| 0x16 | initiate HF connection | UART command (0x17) |
| 0x18 | Enable RX noise reduction when SCO ready | UART command (0x1D) |
| 0x19 | Disable RX noise reduction when SCO ready | UART command (0x1D) |
| 0x1A | Switch RX noise reduction when SCO ready | UART command (0x1D) |
| 0x1B | Enable TX noise reduction when SCO ready | UART command (0x1D) |
| 0x1C | Disable TX noise reduction when SCO ready | UART command (0x1D) |
| 0x1D | Switch TX noise reduction when SCO ready | UART command (0x1D) |
| 0x30 | increase speaker gain | UART command (0x23) |
| 0x31 | decrease speaker gain | UART command (0x23) |
| 0x32 | Play/Pause music | UART command (0x04) |
| 0x33 | Stop music | UART command (0x04) |
| 0x34 | Next song | UART command (0x04) |
| 0x35 | Previous song | UART command (0x04) |
| 0x36 | Fast forward | UART command (0x04) |
| 0x37 | Rewind | UART command (0x04) |
| 0x38 | EQ mode up | UART command (0x1C) |
| 0x39 | EQ mode down | UART command (0x1C) |
| 0x3E | Toggle 3D effect | TBD |
| 0x50 | enter pairing mode (from power off state) | Similar with MMI action 0x5D |
| 0x6C | link last device | UART command (0x17) |
| 0x6D | disconnect all link | UART command (0x18) |
| 0x6E | OHS event 1 | Project related command |
| 0x6F | OHS event 2 | Project related command |
| 0x70 | OHS event 3 | Project related command |
| 0x71 | OHS event 4 | Project related command |
| 0x79 | force buzzer alarm | TBD |
| 0x7B | OHS event 5 | Project related command |
| 0x7C | OHS event 6 | Project related command |
| 0xC0 | OHS event 7 (for Unlimiter project) | Project related command |
| 0xC1 | OHS event 8 (for Unlimiter project) | Project related command |
| 0xC2 | OHS event 9 (for Unlimiter project) | Project related command |
| 0xC3 | OHS event 10 (for Unlimiter project) | Project related command |
| 0xC4 | OHS event 11 (for Unlimiter project) | Project related command |
| 0xC5 | OHS event 12 (for Unlimiter project) | Project related command |
| 0xC6 | OHS event 13 (for Unlimiter project) | Project related command |
| 0xC7 | OHS event 14 (for Unlimiter project) | Project related command |
| 0xC8 | OHS event 15 (for Unlimiter project) | Project related command |
| 0xC9 | OHS event 16 (for Unlimiter project) | Project related command |
| 0xCA | OHS event 17 (for Unlimiter project) | Project related command |
| 0xCB | Switch DSP mode 2 or 3 (for Unlimiter project) | Project related command |
| 0xEA | NSPK Master exit Aux-in SBC Encoder mode | TBD |
| 0xF0 | NSPK MASTER PAGE SLAVE FOR NEW SLAVE | TBD |
| 0xF1 | NSPK SLAVE ENABLE PAGE SCAN FOR NEW MASTER | TBD |
| 0xF2 | NSPK SLAVE USE SLOW PAGE SCAN | TBD |
| 0xF3 | NSPK SLAVE USE FAST PAGE SCAN | TBD |

# References

[1] [3GPP 27.007 v6.8.0 now supersedes and replaces ETS 300 916, “Digital cellular telecommunications system (Phase 2+); AT command set for GSM Mobile Equipment (ME) (GSM 07.07 version 7.5.0)”](http://www.3gpp.org/ftp/Specs/html-info/27007.htm)

[2] [Apple Notification Center Service v1.1 document](https://developer.apple.com/library/ios/documentation/CoreBluetooth/Reference/AppleNotificationCenterServiceSpecification/Introduction/Introduction.html)

[3] HFP 1.5\_SPEC

[4] AVRC 1.3\_SPEC

# Terms and Definitions

|  |  |
| --- | --- |
| AEC | Acoustic Echo Cancellation |
| ANCS | Apple Notification Center Service |
| A/V | A/V Audio/Video |
| AV/C | AV/C The AV/C Digital Interface Command Set |
| AVCTP | AVCTP Audio/Video Control Transport Protocol |
| AVRCP | AVRCP Audio/Video Remote Control Profile |
| EQ | Equalization |
| NR | Noise Reduction |
| NS | Noise Suppression |
|  |  |