Bluetooth Baseband LSI

Panasonic PAN1026

Toshiba TC35661

Extension HCI command document

August.2013

- PANASONIC is continually working to improve the quality and reliability of its products. Nevertheless, semiconductor devices in general can malfunction or fail due to their inherent electrical sensitivity and vulnerability to physical stress. It is the responsibility of the buyer, when utilizing PANASONIC products, to comply with the standards of safety in making a safe design for the entire system, and to avoid situations in which a malfunction or failure of such PANASONIC products could cause loss of human life, bodily injury or damage to property.
  In developing your designs, please ensure that PANASONIC products are used within specified operating ranges as set forth in the most recent PANASONIC products specifications.
- The PANASONIC products listed in this document are intended for usage in general electronics applications (computer, personal equipment, office equipment, measuring equipment, industrial robotics, domestic appliances, etc.). These PANASONIC products are neither intended nor warranted for usage in equipment that requires extraordinarily high quality and/or reliability or a malfunction or failure of which may cause loss of human life or bodily injury ("Unintended Usage"). Unintended Usage include atomic energy control instruments, airplane or spaceship instruments, transportation instruments, traffic signal instruments, combustion control instruments, medical instruments, all types of safety devices, etc.. Unintended Usage of PANASONIC products listed in this document shall be made at the customer's own risk.
- The products described in this document are subject to the foreign exchange and foreign trade laws.
- The information contained herein is presented only as a guide for the applications of our products. No responsibility is assumed by PANASONIC CORPORATION for any infringements of intellectual property or other rights of the third parties which may result from its use. No license is granted by implication or otherwise under any intellectual property or other rights of PANASONIC CORPORATION or others.
- The information contained herein is subject to change without notice.
- The information contained herein is presented only as a guide for the product operation, its functions, and applications. We request that the operation of any application system incorporating this product is fully tested by system vendor.

Revision History

Date	Modification	Note
24th-June-2013	1st Release Based on TC35661APL_ROM203_Extension_HCI_Command_E_5thJune2013	
7th-August-2013	New additional sections are as follows. 3.4.9 Example for patch information data (Former data of SWAP) write with M2 SET command 3.4.10 Example for patch program data write with M2 SET command 3.4.11 Example for patch control(enable/disable) with M2 SET command	

PANASONIC Bluetooth Module PAN1026 CMD(Extension HCI)

Contents

PANASONIC Bluetooth Module PAN1026 CMD(Extension HCI)

1. General

This document is for TOSHIBA extension HCI command, which is used to

(1) Measure RF characteristics such as RADIO TEST

(2) HW and firmware setting

(3) get HW or firmware information

(4) HCI command by Bluetooth SIG

1.1 Supported command

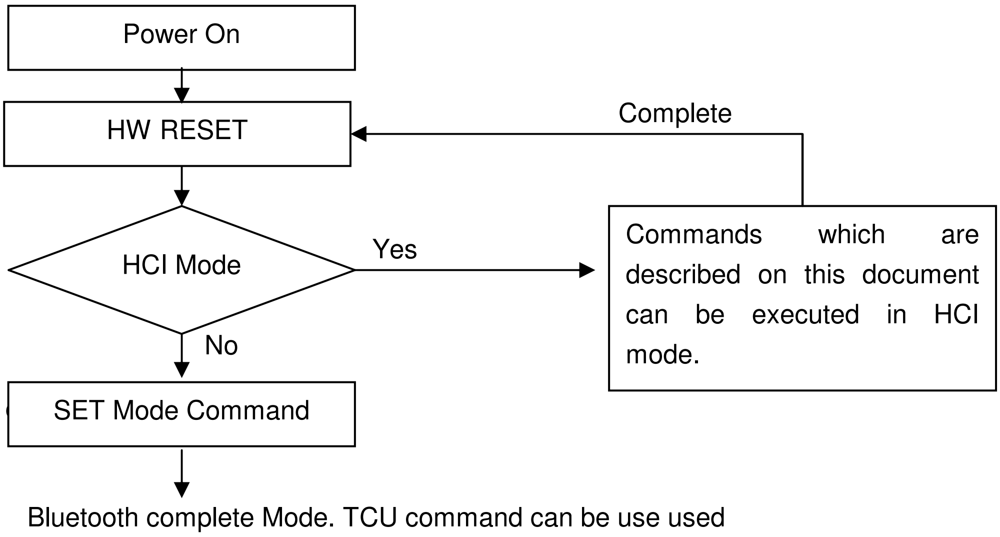
Function	Description
Non modulated carrier generating	For measure RF characteristics
TX bust generation	Modulated signal generation without Bluetooth connection.
Successive scan mode	
Write Bluetooth Address	
Mode change	Mode change from HCI mode to Complete mode
UART Baudrate setting	Default is 115.2kbps. Note: If other baudrate is needed, contact us.
Firmware version	
Bluetooth test mode	This command is for connection to Bluetooth Tester.

1.1.1 Initial Control Sequence

This chapter explains the initial control sequence for TC35661.

After to release Reset sequence, TC35661 is set to HCI mode, Commands which are described on this document can be executed in HCI mode.

To change the complete mode from HCI mode, Host CPU sends the command “”HCI_Set_Mode command” in HCI Vendor Specific command.


Power On
HW RESET
Complete
HCI Mode
Yes
Commands which are described on this document can be executed in HCI mode.
No
SET Mode Command
Bluetooth complete Mode. TCU command can be use used

HW RESET is needed to change from

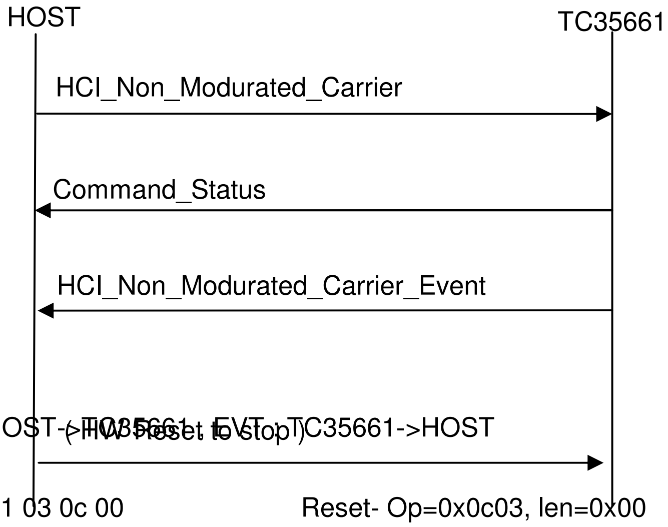
2. Non modulated carrier and TX bust generation for Radio Test

This section shows commands sequence to measure RF characteristics for RADIO TEST.

2.1 Operation for Bluetooth Core Spec. Ver2.1

(1)None modulated carrier (2441MHz)
(2)DH5 TX bust generation (Hopping)
(3)DH5 TX bust generation(2441MHz)
(4)Receiving mode.

2.1.1 Non modulated carrier (2441MHz)



CMD : HOST->TC35661, EVT : TC35661->HOST

(CMD) 01 03 0c 00	Reset- Op=0x0c03, len=0x00
(EVT) 04 0e 04 04 03 0c 00	Command_Complete- NHCP=0x04, Op=0x0c03(Reset), st=0x00(SUCCESS)

(CMD) 01 03 fc 04 00 0b 1c 29	ExCh3-Op=0xfc03,len=0x04
	mesgID=0x0b(LOC_OPERATION_MODE_SET),
	Operation_Mode=0x1c(NON_MODURATED_CARRIER),
	Frequency=0x29(2441MHz)

(EVT) 04 0f 04 00 04 03 fc	Command_Status-st=0x00(SUCCESS),NHCP=0x04,
	Op=0xfc03(ExCh3),

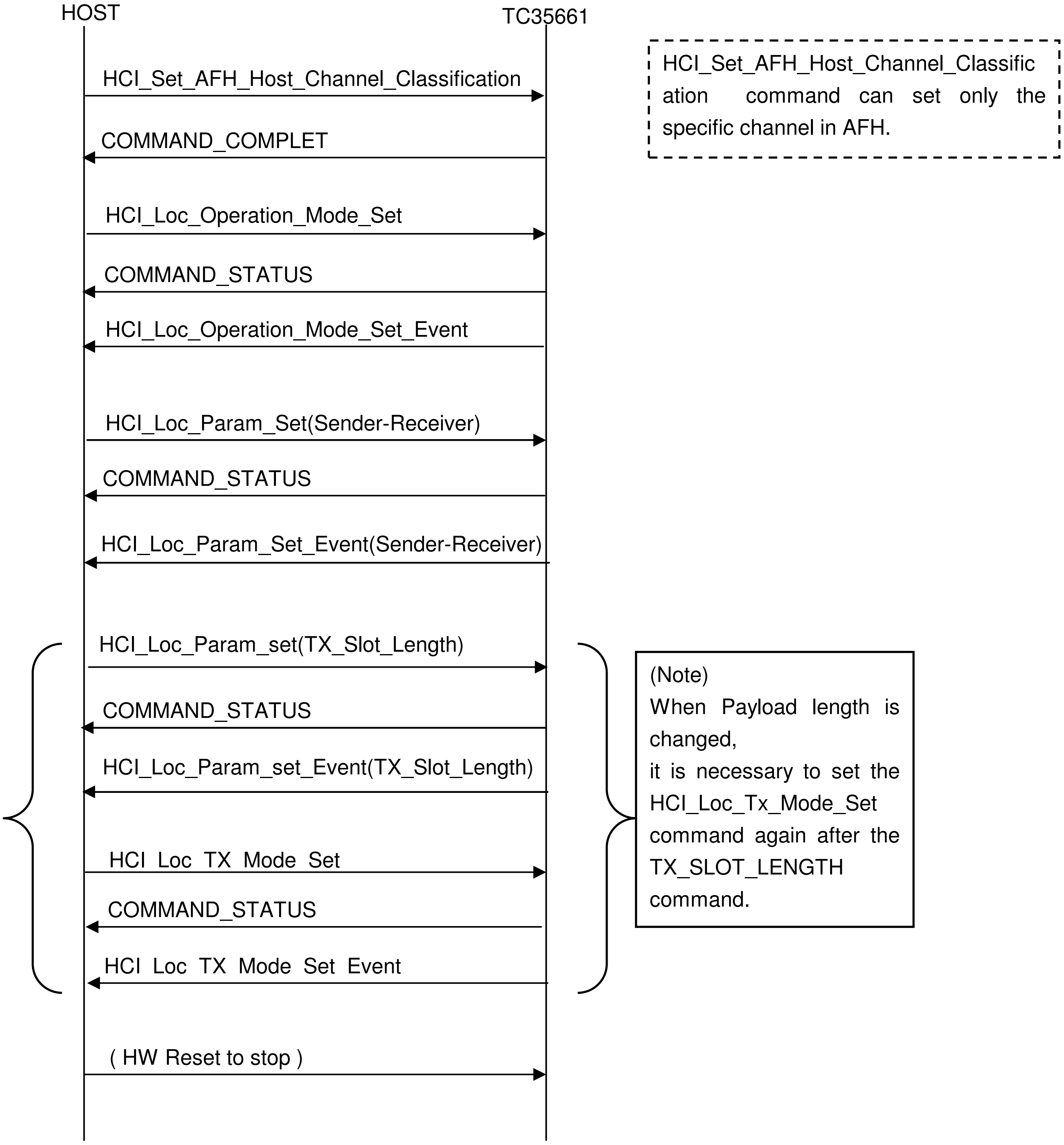
(EVT) 04 ff 05 03 00 0b 00 1c	Ext_Event-ExCh3(LCI),
	MesgID=0x0b(LOC_OPERATION_MODE_SET),st=0x00(LOC_OK),
	Operation_Mode=0x1c(NON_MODURATED_CARRIER)

To stop this test, HW Reset is needed.

2.1.2 TX bust generation for DH5 (Hopping enable)

The following flow shows DH5 TX bust generation (Hopping) setting.

When only a specific channel in AFH is selected, HCI_Set_AFH_Host_Channel_Classification command after HCI_RESET is used.


HOST
TC35661
HCI_Set_AFH_Host_Channel_Classification
COMMAND_COMPLET
HCI_Loc_Operation_Mode_Set
COMMAND_STATUS
HCI_Loc_Operation_Mode_Set_Event
HCI_Loc_Param_Set(Sender-Receiver)
COMMAND_STATUS
HCI_Loc_Param_Set_Event(Sender-Receiver)
HCI_Loc_Param_set(TX_Slot_Length)
COMMAND_STATUS
HCI_Loc_Param_set_Event(TX_Slot_Length)
HCI Loc TX Mode Set
COMMAND_STATUS
HCI Loc TX Mode Set Event
(HW Reset to stop)

HCI_Set_AFH_Host_Channel_Classification command can set only the specific channel in AFH.

(Note)
When Payload length is changed,
it is necessary to set the HCI_Loc_Tx_Mode_Set command again after the TX_SLOT_LENGTH command.

UART log is as follows.

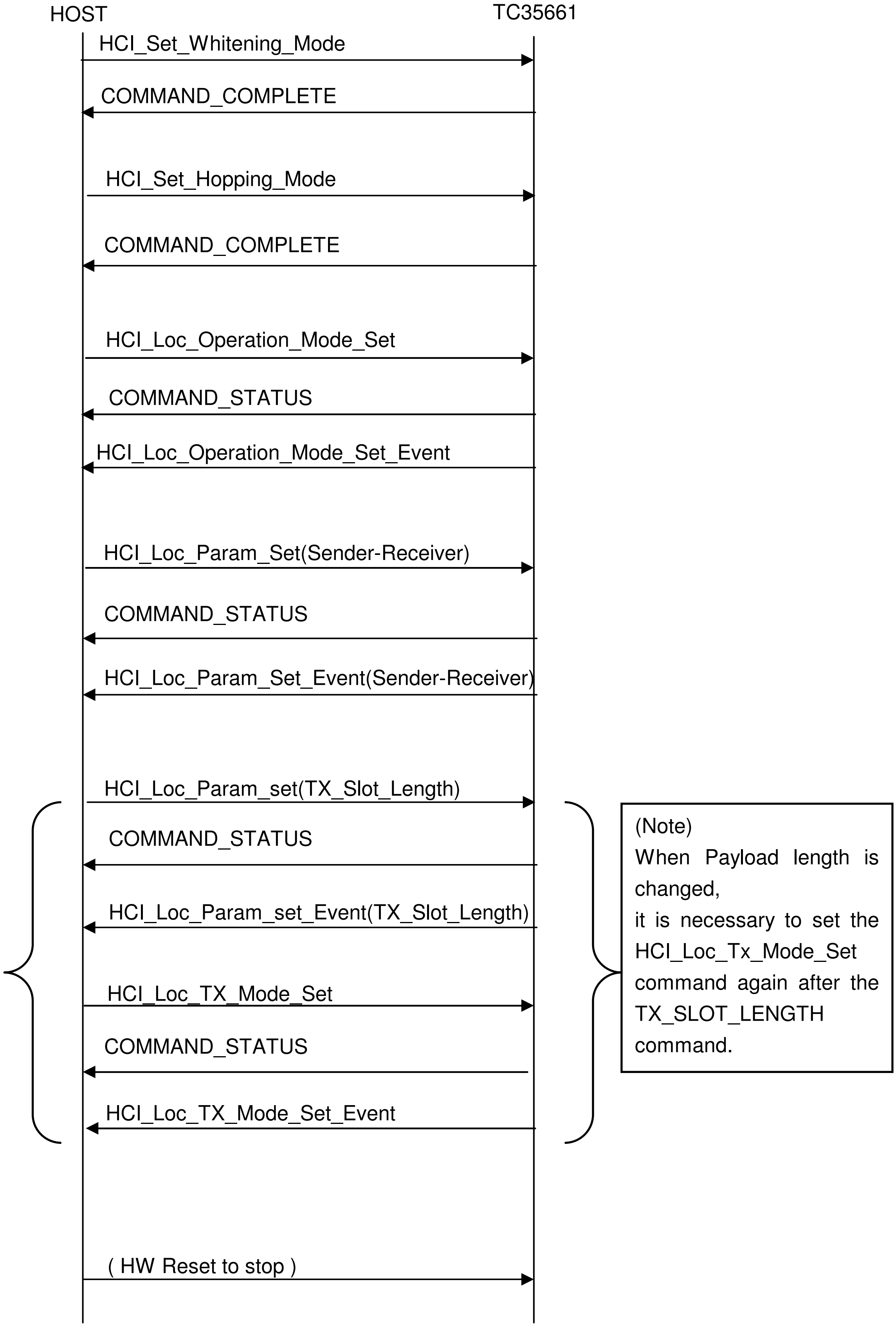
CMD : HOST->TC35661 , EVT : TC35661->HOST

This command set used hopping channel.

```
(CMD) 01 3f 0c 0a ff ff 0f 00 00 00 00 00 00 00
                                    Set_AFH_Host_Channel_Classification- Op=0x0c3f len=0x0a
                                    AFHHostChClass=0x000000000000000fffff（channel 0-19 are used）
(EVT) 04 0e 04 04 3f 0c 00      Command_Complete- NHCP=0x04, Op=0x0c3f, st=0x00(SUCCESS)

(CMD) 01 03 fc 03 00 0b 1b          ExCh3-Op=0xfc03,len=0x03
                                    mesgID=0x0b(LOC_OPERATION_MODE_SET)
                                    Operation_Mode=0x1b(LOCAL)
(EVT) 04 0f 04 00 04 03 fc          Command_Status-st=0x00(SUCCESS),NHCP=0x04
                                    Op=0xfc03(ExCh3)
(EVT) 04 ff 05 03 00 0b 00 1b       Ext_Event-ExCh3(LCI)
                                    MesgID=0x0b(LOC_OPERATION_MODE_SET),st=0x00(LOC_OK)
                                    Operation_Mode=0x1b(LOCAL)

(CMD) 01 03 fc 05 00 13 25 00 00            ExCh3-Op=0xfc03,len=0x05
                                            mesgID=0x13(LOC_PARAM_SET)
                                            Type=0x25(SENDER_RECEIVER) Value=0x0000
(EVT) 04 0f 04 00 04 03 fc                  Command_Status-st=0x00(SUCCESS),NHCP=0x04
                                            Op=0xfc03(ExCh3)
(EVT) 04 ff 06 03 00 13 00 25 00            Ext_Event-ExCh3(LCI),MesgID=0x13(LOC_PARAM_SET)
                                            st=0x00(LOC_OK), Type=0x25(SENDER_RECEIVER)
                                            Value=0x0900

(CMD) 01 03 fc 05 00 13 22 00 05            ExCh3-Op=0xfc03,len=0x05
                                            mesgID=0x13(LOC_PARAM_SET)
                                            Type=0x22(TX_SLOT_LENGTH) Value=0x0500
(EVT) 04 0f 04 00 04 03 fc                  Command_Status-st=0x00(SUCCESS),NHCP=0x04
                                            Op=0xfc03(ExCh3)
(EVT) 04 ff 06 03 00 13 00 22 00            Ext_Event-ExCh3(LCI),MesgID=0x13(LOC_PARAM_SET)
                                            st=0x00(LOC_OK), Type=0x22(TX_SLOT_LENGTH)
                                            Value=0xf400

(CMD) 01 03 fc 04 00 08 17 09               ExCh3-Op=0xfc03,len=0x04
                                            mesgID=0x08(LOC_TX_MODE_SET)
                                            Data_Type=0x17(BT_DATA_PRBS9)
                                            TX_Mode=0x09(BT_ON)
(EVT) 04 0f 04 00 04 03 fc                  Command_Status-st=0x00(SUCCESS),NHCP=0x04
                                            Op=0xfc03(ExCh3)
(EVT) 04 ff 06 03 00 08 00 17 09            Ext_Event-ExCh3(LCI)
                                            MesgID=0x08(LOC_TX_MODE_SET),st=0x00(LOC_OK)
                                            Data_Type=0x17(BT_DATA_PRBS9)
                                            TX_Mode=0x09(BT_ON)
```

To stop this test, HW Reset is needed.

2.1.3 TX burst generation for DH5 (Fix frequency :2441MHz)


HOST
TC35661
HCI_Set_Whitening_Mode
COMMAND_COMPLETE
HCI_Set_Hopping_Mode
COMMAND_COMPLETE
HCI_Loc_Operation_Mode_Set
COMMAND_STATUS
HCI_Loc_Operation_Mode_Set_Event
HCI_Loc_Param_Set(Sender-Receiver)
COMMAND_STATUS
HCI_Loc_Param_Set_Event(Sender-Receiver)
HCI_Loc_Param_set(TX_Slot_Length)
COMMAND_STATUS
HCI_Loc_Param_set_Event(TX_Slot_Length)
HCI_Loc_TX_Mode_Set
COMMAND_STATUS
HCI_Loc_TX_Mode_Set_Event
(HW Reset to stop)

(Note)
When Payload length is changed,
it is necessary to set the HCI_Loc_Tx_Mode_Set command again after the TX_SLOT_LENGTH command.

UART log is as follows.

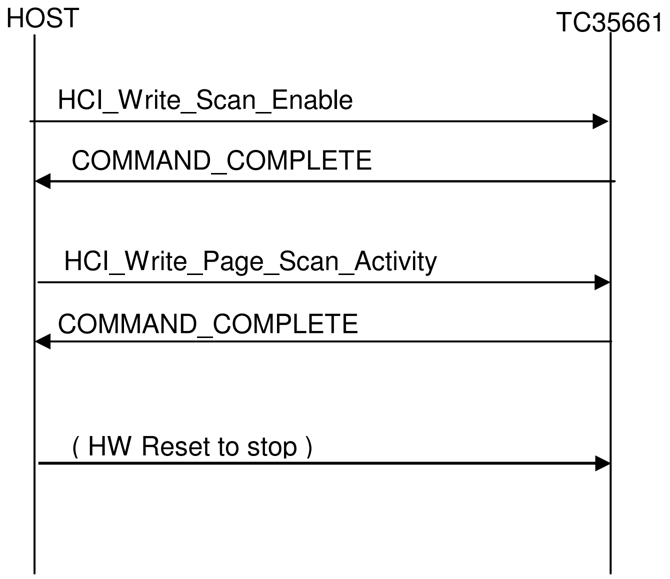
CMD : HOST->TC35661 , EVT : TC35661->HOST

```
(CMD) 01 08 fc 03 00 21 00    ExCh8-Op=0xfc08,len=0x03,ExOp=0x21(Set_Whitening_Mode)
                              Whitening_Flag=0x00(ON)
(EVT) 04 0e 05 04 08 fc 00 21 Command_Complete-NHCP=0x04,Op=0xfc08(ExCh8)
                              st=0x00(SUCCESS), ExOp=0x21(Set_Whitening_Mode)

(CMD) 01 08 fc 04 00 20 01 29 ExCh8-Op=0xfc08,len=0x04
                                      ExOp=0x20(Set_Hopping_Mode),Hopping_Flag=0x01(OFF)
                                      Frequency=0x29(2441MHz)
(EVT) 04 0e 05 04 08 fc 00 20          Command_Complete-NHCP=0x04,Op=0xfc08(ExCh8)
                                      st=0x00(SUCCESS) ExOp=0x20(Set_Hopping_Mode)

(CMD) 01 03 fc 03 00 0b 1b             ExCh3-Op=0xfc03,len=0x03
                                      mesgID=0x0b(LOC_OPERATION_MODE_SET)
                                      Operation_Mode=0x1b(LOCAL)
(EVT) 04 0f 04 00 04 03 fc             Command_Status-st=0x00(SUCCESS), NHCP=0x04
                                      Op=0xfc03(ExCh3)
(EVT) 04 ff 05 03 00 0b 00 1b          Ext_Event-ExCh3(LCI)
                                      MesgID=0x0b(LOC_OPERATION_MODE_SET)
                                      st=0x00(LOC_OK) Operation_Mode=0x1b(LOCAL)

(CMD) 01 03 fc 05 00 13 25 00 00       ExCh3-Op=0xfc03,len=0x05
                                      mesgID=0x13(LOC_PARAM_SET)
                                      Type=0x25(SENDER_RECEIVER) Value=0x0000
(EVT) 04 0f 04 00 04 03 fc             Command_Status-st=0x00(SUCCESS),NHCP=0x04
                                      Op=0xfc03(ExCh3)
(EVT) 04 ff 06 03 00 13 00 25 00       Ext_Event-ExCh3(LCI),MesgID=0x13(LOC_PARAM_SET)
                                      st=0x00(LOC_OK), Type=0x25(SENDER_RECEIVER)
                                      Value=0x0000

(CMD) 01 03 fc 05 00 13 22 00 05       ExCh3-Op=0xfc03,len=0x05
                                      mesgID=0x13(LOC_PARAM_SET)
                                      Type=0x22(TX_SLOT_LENGTH) Value=0x0500
(EVT) 04 0f 04 00 04 03 fc             Command_Status-st=0x00(SUCCESS),NHCP=0x04
                                      Op=0xfc03(ExCh3)
(EVT) 04 ff 06 03 00 13 00 22 00       Ext_Event-ExCh3(LCI),MesgID=0x13(LOC_PARAM_SET)
                          st=0x00(LOC_OK),Type=0x22(TX_SLOT_LENGTH),Value=0x0000

(CMD) 01 03 fc 04 00 08 17 09 ExCh3-Op=0xfc03,len=0x04
                                      mesgID=0x08(LOC_TX_MODE_SET)
                                      Data_Type=0x17(BT_DATA_PRBS9)
                                      TX_Mode=0x09(BT_ON)
(EVT) 04 0f 04 00 04 03 fc             Command_Status-st=0x00(SUCCESS),NHCP=0x04
                                       Op=0xfc03(ExCh3)
(EVT) 04 ff 06 03 00 08 00 17 09       Ext_Event-ExCh3(LCI)
                                      MesgID=0x08(LOC_TX_MODE_SET)
                                      st=0x00(LOC_OK),Data_Type=0x17(BT_DATA_PRBS9)
```

TX_Mode=0x09(BT_ON)

To stop this test, HW Reset is needed.

2.1.4 Successive receiving mode (Hopping ON)

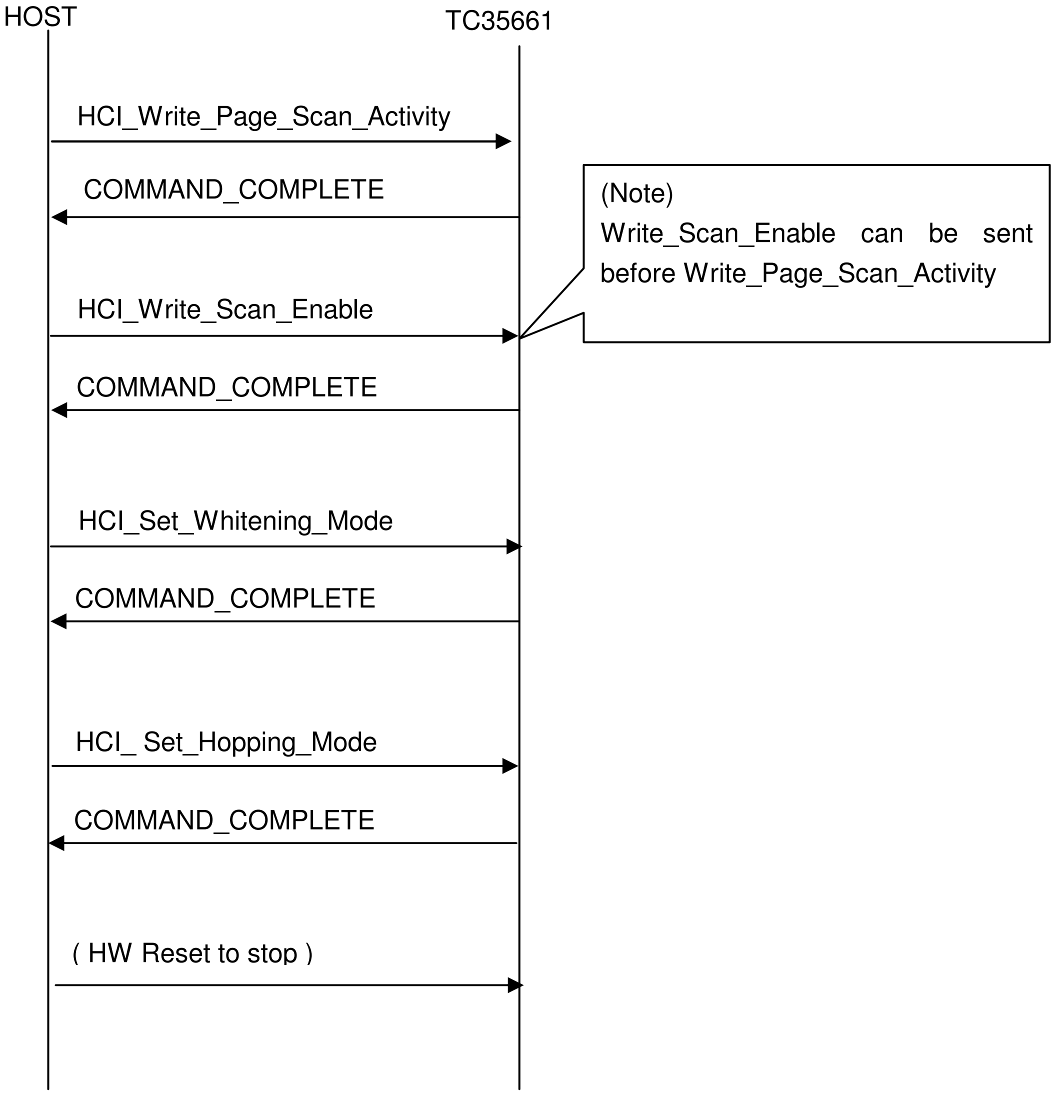


UART log is as follows.
CMD: HOST->TC35661, EVT: TC35661->HOST

```
(CMD) 01 1a 0c 01 02          Write_Scan_Enable- Op=0x0c1a len=0x01 Enable=0x02(Page Scan)
(EVT) 04 0e 04 04 1a 0c 00    Command_Complete-NHCP=0x04, Op=0x0c1a(Write_Scan_Enable)
                              st=0x00(SUCCESS)

(CMD) 01 1c 0c 04 00 08 00 08 Write_Page_Scan_Activity-Op=0x0c1c,len=0x04
                                  Page_Scan_Interval=0x0800 Page_Scan_Window=0x0800
(EVT) 04 0e 04 04 1c 0c 00            Command_Complete-NHCP=0x04,
                                 Op=0x0c1c(Write_Page_Scan_Activity) st=0x00(SUCCESS)
```

To stop this test, HW Reset is needed.

2.1.5 Successive receiving mode (Hopping OFF)



CMD:HOST->TC35661 EVT:TC35661->HOST

```
(CMD) 01 1c 0c 04 00 08 00 08 Write_Page_Scan_Activity-Op=0x0c1c,len=0x04
                                    Page_Scan_Interval=0x0800 Page_Scan_Window=0x0800
(EVT) 04 0e 04 04 1c 0c 00            Command_Complete-NHCP=0x04,
                                   Op=0x0c1c(Write_Page_Scan_Activity) st=0x00(SUCCESS)

(CMD) 01 1a 0c 01 02          Write_Scan_Enable- Op=0x0c1a len=0x01 Enable=0x02(Page Scan)
(EVT) 04 0e 04 04 1a 0c 00    Command_Complete-NHCP=0x04, Op=0x0c1a(Write_Scan_Enable)
                              st=0x00(SUCCESS)

(CMD) 01 08 fc 03 00 21 00    ExCh8-Op=0xfc08,len=0x03,ExOp=0x21(Set_Whitening_Mode)
                              Whitening_Flag=0x00(ON)
(EVT) 04 0e 05 04 08 fc 00 21 Command_Complete-NHCP=0x04,Op=0xfc08(ExCh8)
                              st=0x00(SUCCESS), ExOp=0x21(Set_Whitening_Mode)

(CMD) 01 08 fc 04 00 20 01 29 ExCh8-Op=0xfc08,len=0x04
                                    ExOp=0x20(Set_Hopping_Mode),Hopping_Flag=0x01(OFF)
                                    Frequency=0x29(2441MHz)
(EVT) 04 0e 05 04 08 fc 00 20        Command_Complete-NHCP=0x04,Op=0xfc08(ExCh8)
                                    st=0x00(SUCCESS) ExOp=0x20(Set_Hopping_Mode)
```

To stop this test, HW Reset is needed.

3. Command format

3.1 Packet Format in HCI mode

UART Protocol in HCI mode is based on Bluetooth Core Spec.H4(UART Transport Layer).

The HCI packet indicator shall be sent immediately before the HCI packet.

HCI packet type	HCI packet indicator
HCI Command Packet	0x01
HCI ACL Data Packet	0x02
HCI Synchronous Data Packet	0x03 (No Support)
HCI Event Packet	0x04

3.2 Vendor Specific Command Explanation

3.2.1 < HCI_LOC_OPERATION_MODE_SET >

To set local mode in order to permit changing RF parameters
RF parameter can be changed during local mode.
HW reset is needed to change from local mode to normal mode.

HCI_LOC_OPERATION_MODE_SET

	Setting value	content
Byte 0	03	OCF
Byte 1	FC	OGF+OCF
Byte 2	03	Command length
Byte 3	00	Reserved
Byte 4	0B	LOC_OPERATION_MODE_SET command
Byte 5	1B	Set Local mode

HCI_LOC_OPERATION_MODE_SET Event

	Setting value	content
Byte 0	FF	Event code
Byte 1	05	Command length
Byte 2	03	OCF
Byte 3	00	Reserved
Byte 4	0B	LOC_OPERATION_MODE_SET command
Byte 5	XX	status 00: Success 01: Fault
Byte 6	1B	1B: Local

3.2.2 < HCI_SET_HOPPING_MODE >

To set enable/disable Hopping sequence.
When hopping is disabled, frequency channel setting is needed to at the same time.
Default value is Hopping.

HCI_SET_HOPPING_MODE

	Setting value	Content
Byte 0	08	OCF
Byte 1	FC	OGF+OCF
Byte 2	04	Command length
Byte 3	00	Reserved
Byte 4	20	HCI_SET_HOPPING_MODE command
Byte 5	XX	Hopping mode 01: Disable 00: Enable
Byte 6	XX	Frequency setting in case of Hopping disable. 2400MHz is set to 00. 1MHz is increased, then add “1” value. Example, In case of 2440MHz, This value should be set 40(0x28)

HCI_SET_HOPPING_MODE Event

	Setting value	Content
Byte 0	0E	Event code
Byte 1	05	Command length
Byte 2	04	OCF
Byte 3	08	OCF
Byte 4	FC	OCF
Byte 5	XX	Status 00: Success 01: Fault
Byte 6	20	HCI_SET_HOPPING_MODE command

3.2.3 < HCI_LOC_SET_WHITENING_MODE >

To set enable/disable whitening. Default value is enabling.

HCI_LOC_SET_WHITENING_MODE

	Setting value	Content
Byte 0	08	OCF
Byte 1	FC	OGF+OCF
Byte 2	03	Command length
Byte 3	00	Reserved
Byte 4	21	HCI_LOC_SET_WHITENING_MODE command
Byte 5	XX	Whitening mode 01: Disable 00: Enable

HCI_LOC_SET_WHITENING_MODE Event

	Setting value	content
Byte 0	0E	Event code
Byte 1	05	Command length
Byte 2	04	OCF
Byte 3	08	OCF
Byte 4	FC	OCF
Byte 5	XX	status 00: Success 01: Fault
Byte 6	21	HCI_LOC_SET_WHITENING_MODE command

3.2.4 < HCI_NON_MODURATED_CAREER >

To generate non modulated carrier. Hopping is always disabled.
HCI_RESET is used to stop generation.

HCI_NON_MODURATED_CAREER

	Setting value	Content
Byte 0	03	OCF
Byte 1	FC	OGF+OCF
Byte 2	04	Command length
Byte 3	00	Reserved
Byte 4	0B	HCI_NON_MODURATED_CAREER command
Byte 5	1C	Fixed value
Byte 6	XX	Frequency setting 2400MHz is set to 00. 1MHz is increased, then add “1” value. Example, In case of 2440MHz, This value should be set 40(0x28)

HCI_NON_MODURATED_CAREER Event

	Setting value	Content
Byte 0	FF	Event code
Byte 1	05	Command length
Byte 2	03	OCF
Byte 3	00	Reserved
Byte 4	0B	HCI_NON_MODURATED_CAREER command
Byte 5	XX	status 00: Success 01: Fault
Byte 6	1C	Fixed value

3.2.5 < HCI_LOC_TX_MODE_SET >

To set TX burst generation. Payload length should be set in advance with HCI_LOC_TX_SLOT_LENGTH command.
This command operates only TX slot. RX slot is not operated in its state
LOCAL_OPERATION_MODE and HCI_ SENDER_RECEIVE command are needed in advance.
TX burst generation sequence is descried on section 2.1.2.

HCI_LOC_TX_MODE_SET

	Setting value	Content
Byte 0	03	OCF
Byte 1	FC	OGF+OCF
Byte 2	04	Command length
Byte 3	00	Reserved
Byte 4	08	HCI_LOC_TX_MODE_SET command
Byte 5	XX	Payload content 13: 11111111 14: 00000000 15: 0101010101 17: PRBS9 33: 00001111
Byte 6	09	BT_ON

Command_Status is transmitted.
Refer to the following event.

HCI_LOC_TX_MODE_SET event

	Setting vale	Content
Byte 0	FF	Event code
Byte 1	06	Command length
Byte 2	03	OCF
Byte 3	00	Reserved
Byte 4	08	HCI_LOC_TX_MODE_SET command
Byte 5	XX	status 00: Success 01: Fault
Byte 6	XX	Set value
Byte 7	09	BT_ON

3.2.6 < HCI_LOC_TX_SLOT_LENGTH >

To set payload length for TX burst generation.
This command is valid during TX burst generation and local mode.
HCI_LOC_OPERATION_MODE_SET command can set local mode.

HCI_LOC_TX_SLOT_LENGTH

	Setting value	Content
Byte 0	03	OCF
Byte 1	FC	OGF+OCF
Byte 2	05	Command length
Byte 3	00	Reserved
Byte 4	13	LOC_PARAM_SET Command
Byte 5	22	HCI_LOC_TX_SLOT_LENGTH Command
Byte 6	00	Always 0x00
	XX	Payload length 05: DH5 03: DH3 01: DH1 25: 2-DH5 23: 2-DH3 21: 2-DH1 35: 3-DH5 33: 3-DH3 31: 3-DH1

Command_Status is transmitted.
Refer to the following event.

HCI_LOC_TX_SLOT_LENGTH event

	Setting value	Content
Byte 0	FF	Event code
Byte 1	06	Command length
Byte 2	03	OCF
Byte 3	00	Reserved
Byte 4	13	LOC_PARAM_SET Command
Byte 5	XX	status 00: Success 01: Fault
Byte 6	22	HCI_LOC_TX_SLOT_LENGTH Command
Byte 7	00	Reserved

3.2.7 < HCI_LOC_SENDER_RECEIVER >

This command should be set before HCI_LOC_TX_MODE_SET command.
This command is valid during local mode.
HCI_LOC_OPERATION_MODE_SET command can set local mode**.**
Moreover, this command is effective only when LOCAL_OPERATION mode is enabling.

HCI_LOC_SENDER_RECEIVER

	Setting value	Content
Byte 0	03	OCF
Byte 1	FC	OGF+OCF
Byte 2	05	Command length
Byte 3	00	Reserved
Byte 4	13	LOC_PARAM_SET Command
Byte 5	25	SENDER_RECEIVER Command
Byte 6	0000	TX burst generation setting Except 0 is not available.

HCI_LOC_SENDER_RECEIVER event

	Setting value	Content
Byte 0	FF	Event code
Byte 1	06	Command length
Byte 2	03	OCF
Byte 3	00	Reserved
Byte 4	13	LOC_PARAM_SET Command
Byte 5	XX	status 00: Success 01: Fault
Byte 6	25	HCI_LOC_SENDER_RECEIVER Command
Byte 7	00	Reserved

3.2.8 < HCI_WRITE_BD_ADDR >

To set BD_ADDR to the module
This command should be used while No Connection state and Scan Disable state.

HCI_WRITE_BD_ADDR

	Setting value	Content
Byte 0	13	OCF
Byte 1	10	OGF＋OCF
Byte 2	06	Command length
Byte 3	XX	BD_ADDR(LSB)
Byte 4	XX	BD_ADDR(LSB+1)
Byte 5	XX	BD_ADDR(LSB+2)
Byte 6	XX	BD_ADDR(LSB+3)
Byte 7	XX	BD_ADDR(LSB+4)
Byte 8	XX	BD_ADDR(MSB)

HCI_WRITE_BD_ADDR event

	Setting value	Content
Byte 0	0E	Event code (Command Complete Event)
Byte 1	04	Command length
Byte 2	xx	Num HCI Command Packets
Byte 3	13	OCF
Byte 4	10	OGF＋OCF
Byte 5	xx	Status 00: Success Except-0: Fail

3.2.9 < HCI_SET_MODE >

To change from HCI mode to complete mode.
HW RESET is needed to change from complete mode to HCI mode.

HCI_SET_MODE

	Setting value	Content
Byte 0	08	OCF
Byte 1	FC	OGF＋OCF
Byte 2	03	Command length
Byte 3	00	Reserved
Byte 4	99	SET MODE command
Byte 5	01	Fixed value

HCI_SET_MODE event

	Setting value	Content
Byte 0	FF	Event code
Byte 1	05	Command length
Byte 2	08	OCF
Byte 3	00	Reserved
Byte 4	99	SET MODE command
Byte 5	XX	Status 00: Success Except-0: Fail
Byte 6	01	Fixed value

3.2.10 < HCI_UART_RTSCTS_Control >

To Control UART RTS/CTS control. This command is used to control GPIO by Host CPU.
This command execution is needed before GIOP control M2 command.

HCI_UART_RTSCTS_Control

	Setting value	Content
Byte 0	08	OCF
Byte 1	FC	OGF＋OCF
Byte 2	03	Command length
Byte 3	00	Reserved
Byte 4	93	RTS/CTS setting command
Byte 5	XX	Enable/Disable setting. FF: Enable (Initial value) 00: Disable

HCI_UART_RTSCTS_Control event

	Setting value	Content
Byte 0	FF	Event code
Byte 1	05	Command length
Byte 2	08	OCF
Byte 3	00	Reserved
Byte 4	93	RTS/CTS setting command
Byte 5	XX	Status 00: Success Except-0: Fail
Byte 6	XX	Enable/Disable setting value.

3.3 HCI Command Explanation

3.3.1 <HW Error Event>

Notify the error generation of UART.
Although this event is a standard event of Bluetooth, since the parameter is peculiar to a vendor,
it is described on this document.
If this error occurs, the command or data received immediately before are released.
Therefore, Host CPU needs to send against command without response. As resending timing, after this event generating, resend after waiting 10 ms or more.

HW error event

	Setting value	Content
Byte 0	10	Event code
Byte 1	01	Command length
Byte 2	XX	**Errors** 20: Short of receiving packet. Timer of maximum transmit interval between each bytes is expired. Timer value is 5ms fixed. If this error code occurs, check the transmitting Byte interval from HOST. 21: Stop bit error It is generated, if the clock deviation of UART is large. Please check the deviation of a clock, if this error code occurs. SPEC of our company is TDB. 22: Over write error Since data was received from HOST during RTS control (data stop request), overwrite of data has occurred inside. If this error code occurs, please check the flow control of a HOST side.

3.3.2 < Set_AFH_Host_Channel_Classification >

This command selects the specific frequency in AFH.

Set_AFH_Host_Channel_Classification command

	Setting value	Content
Byte 0	3F	OCF
Byte 1	0C	OGF+OCF
Byte 2	0A	Command length
Byte 3	XX	Each channel is expressed by 1Bit. Used channel is 1, can't used channel is 0. Setting between channel from 0 to 7
Byte 4	XX	Setting between channel from 8 to 15
Byte 5	XX	Setting between channel from 16 to 23
Byte 6	XX	Setting between channel from 24 to 31
Byte 7	XX	Setting between channel from 32 to 39
Byte 8	XX	Setting between channel from 40 to 47
Byte 9	XX	Setting between channel from 48 to 55
Byte 10	XX	Setting between channel from 56 to 63
Byte 11	XX	Setting between channel from 64 to 71
Byte 12	XX	Setting between channel from 72 to 78

Set_AFH_Host_Channel_Classification Event

	Setting value	Content
Byte 0	0E	Event code
Byte 1	04	Command length
Byte 2	04	Num_HCI_Command_Packets
Byte 3	3F	Commnad_Opcode
Byte 4	0C	Commnad_Opcode
Byte 5	XX	Status 00: Success Except 00: Fault

3.3.3 < Write_Page_Scan_Activity >

This command will write the values for the Page_Scan_Interval and Page_Scan_Window configuration parameters. The Page_Scan_Window shall be less than or equal to the Page_Scan_Interval.

Write_Page_Scan_Activity

	Setting value	Content
Byte 0	1C	OCF
Byte 1	0C	OGF+OCF
Byte 2	04	Command length
Byte 3	00	Page Scan Interval
Byte 4	08	Page Scan Interval
Byte 5	00	Page Scan Window
Byte 6	08	Page Scan Window

Write_Page_Scan_Activity event

	Setting value	Content
Byte 0	0E	Event code
Byte 1	04	Command length
Byte 2	04	Num_HCI_Command_Packets
Byte 3	1C	Commnad_Opcode
Byte 4	0c	Commnad_Opcode
Byte 5	00	Status 00: Success Except 00: Fault

3.3.4 <Command Status Event >

The Command Status event is used to indicate that the command described by the Command_Opcode parameter has been received, and that the TC35661 is currently performing the task for this command.

Command Status Event

	Setting value	Content
Byte 0	0F	Event code
Byte 1	04	Command length
Byte 2	00	Status 0x00 Success
Byte 3	04	NHCP (Fixed value)
Byte 4	XX	Command Opcode
Byte 5	XX	Command Opcode

3.3.5 < HCI_LOC_DBUS_READ >

This command is used to read data from DBUS (RF interface).

HCI_LOC_DBUS_READ

	Setting value	Content
Byte0	03	OCF
Byte1	FC	OGF＋OCF
Byte2	03	Command length
Byte3	00	Reserved
Byte4	C3	LOC_DBUS_READ
Byte5	XX	DBUS address

HCI_LOC_DBUS_READ event

	Setting value	Content
Byte0	FF	Event code
Byte1	07	Command length
Byte2	03	OCF
Byte3	00	Reserved
Byte4	C3	LOC_DBUS_READ
Byte5	XX	Status 00: Success Except-0: Fail
Byte6	XX	DBUS address
Byte7	XX	Read data from DBUS (LSB)
Byte8	XX	Read data from DBUS (MSB)

3.3.6 < HCI_LOC_DBUS_WRITE >

This command is used for write data from DBUS (RF interface).
The value set by this command is lost when reset or power OFF.

HCI_LOC_DBUS_WRITE

	Setting value	Content
Byte0	03	OCF
Byte1	FC	OGF＋OCF
Byte2	05	Command length
Byte3	00	Reserved
Byte4	C2	LOC_DBUS_WRITE
Byte5	XX	DBUS address
Byte6	XX	Write data to DBUS (LSB)
Byte7	XX	Write data to DBUS (MSB)

HCI_LOC_DBUS_WRITE event

	Setting value	Content
Byte0	FF	Event code
Byte1	04	Command length
Byte2	03	OCF
Byte3	00	Reserved
Byte4	C2	LOC_DBUS_WRITE
Byte5	XX	Status 00: Success Except-0: Fail

TC35661 DBUS Register list

DBUS address	Content
0xA6 (Dev addr 0x05, addr 0x06)	Default: 0x0100 bit 15-12: Reserved (Set to 0) bit 11-4: Fine-tuning for the crystal oscillation frequency (Adjustment of internal capacitor array for OSC_CapTrim) Default: 0x10 Maximum: 0x1F Minimum: 0x00 bit 3-0 Resereved (Set to 0)

3.3.7 < HCI_LOC_WRITE_MEM >

This command is used for write data to the memory in the Chiron.

HCI_LOC_WRITE_MEM

	Setting value	Content
Byte0	03	OCF
Byte1	FC	OGF＋OCF
Byte2	08	Command length
Byte3	00	Reserved
Byte4	D1	LOC_WRITE_MEM
Byte5	XX	Write address (LSB)
Byte6	XX	Write address (LSB+1)
Byte7	XX	Write address (LSB+2)
Byte8	XX	Write address (MSB)
Byte9	XX	Write data (LSB)
Byte10	XX	Write data (MSB)

HCI_LOC_WRITE_MEM event

	Setting value	Content
Byte0	FF	Event code
Byte1	05	Command length
Byte2	03	OCF
Byte3	00	Reserved
Byte4	D1	LOC_WRITE_MEM
Byte5	XX	Status 00: Success Except-0: Fail
Byte6	00	Fixed to 0x00.

3.4 Module Maintenance (M2) Command explanation

3.4.1 < HCI_M2_Message_Set >

This command is used to set HW and Firmware setting or Module control.

M2 Message_Set command

Parameters	Value	Parameter Description
Byte0	08	OCF
Byte1	FC	OGF＋OCF
Byte2	XX	Command length
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	00	Reserved
Byte8	X4	X：Accepter
Byte9	XX	Information ID
Byte10	FF	Reserved (0xFF fixed)
Byte11	XX	Data type 00:No information 01:uint8 02:uint16 03:uint32 04:uint64 81:int8 82:int16 83:int32 84:int64 0F:Strings (0x00 terminated) 10:Byte array(First byte is length)
Byte12-		Data (If Data type is 0x00, Data doesn't exist.)

M2Message_Set event

Parameters	Value	Parameter Description
Byte0	FF	Event code
Byte1	XX	Command length
Byte2	08	OCF
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	00	Reserved
Byte8	X4	X：Accepter
Byte9	XX	Information ID
Byte10	XX	Command result 00：successful 01：Not supported Information ID 02：Not supported Information type 03：Data size exceed byte length when byte array transmit 04：Information data error 05：Voice processing executing 06：Audio processing executing 07：Voice recognition processing executing 08：Resource lack FE：Other errors Others：Error
Byte11	XX	Data type 00：No information data 01：uint8 02：uint16 03：uint32 04：uint64 81：int8 82：int16 83：int32 84：int64 0F：Strings (0x00 terminated) 10：Byte array(First byte is length)
Byte12-		Data (If Data type is 0x00，Data doesn't exist.)

Next Page show example for this command

3.4.2 Example for UART Baudrate setting with M2 SET command

This command changes UART Baudrate.
100ms interval is needed after using baudrate setting command.
Baudrate is changed after event generation.
For example, current baudrate is 9600bps. Baudrate is changed to 115.2kbps.
Baudrate setting event is used 9600bps.

M2_BTL_SET_BAUDRATE command

Parameters	Value	Parameter Description
Byte0	08	OCF
Byte1	FC	OGF＋OCF
Byte2	11	Command length
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	00	Reserved
Byte8	14	1： Accepter
Byte9	42	Information ID=Baudrate setting
Byte10	FF	Reserved (0xFF fixed)
Byte11	10	Data type 10：Byte array(First byte is length)
Byte12	07	Data length
Byte13-14	XXXX	Setting the first division ratio for baudrate. Setting range is 0x0001 - 0xFFFF. * Setting example is as below.
Byte15	XX	Setting the second division ratio for baudrate. Bit3-0 are fixed to 0. Bit7-4 are used for setting the second division ratio. Can not be set other than following values. 1010: 17 division ratio, 1001: 16 division ratio, 1000: 15 division ratio, 0111: 14 division ratio(default), 0110: 13 division ratio, 0101: 12 division ratio * Setting example is as below.
Byte16-19	00000000	Fixed to 0x00000000.

* Setting example from Byte13 to Byte19
[Baudrate = 39MHz÷First division ratio÷Second division ratio]
In case of Baudrate = 921.6kbps
(Byte13) 03 00 70 00 00 00 00 (Byte19)
-> 39MHz÷0x0003÷14 division ratio(0111 0000) = 928571.4286bps
In case of Baudrate = 115.2kbps
(Byte13) 1A 00 60 00 00 00 00 (Byte19)
-> 39MHz÷0x001A÷13 division ratio(0110 0000) = 115384.6154bps
In case of Baudrate = 9600bps
(Byte13) ef 00 a0 00 00 00 00 (Byte19)
-> 39MHz÷0x00EF÷17 division ratio(1010 0000) = 9598.818607bps

M2_BTL_SET_BAUDRATE event

Parameters	Value	Parameter Description
Byte0	FF	Event code
Byte1	XX	Command length
Byte2	08	OCF
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	00	Reserved
Byte8	14	1： Accepter
Byte9	42	Information ID=Baudrate setting
Byte10	00	Command result 00：successful
Byte11	00	Data type 00：No information data

3.4.3 Example for DeepSleep setting with M2 SET command

This command is used to Deep Sleep configuration setting.

M2_BTL_SET_DEEP_SLEEP command

Parameters	Value	Parameter Description
Byte0	08	OCF
Byte1	FC	OGF＋OCF
Byte2	1C	Command length
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	000000	Reserved
Byte8	14	1：Accepter 4：Initiator
Byte9	68	Information ID = BTL_SET_DEEP_SLEEP
Byte10	FF	Reserved. 0xFF fixed.
Byte11	10	Data type 10：Byte array(First byte is length)
Byte12	12	Parameter length. 0x12 fixed.
Byte13	XX	CLKREQ signal output format setting. A high level of CLKREQ pin indicates a request for Oscillator. 00：Work deep-sleep (initial value) 01：Always L (CLKREQ terminal unnecessary) 02-FF：Always H (32KHz)
Byte14-17	XXXXXXXX	Crystal stabilization time (us). Initial setting = 0x00000BB8 (3000us) This value can't be changed by Toshiba permission. This value influence HW operation.
Byte18	XX	Deep-sleep instructions / Set Notify specific interface Bit0：GPIO (0=No notification / 1=Notification) GPIO0:Request ,GPIO1:Notify Bit1：UART (0=No notification / 1=Notification) Bit2：USB (0=No notification / 1=Notification) Bit0 and Bit1 and Bit2 are exclusive. SPP complete firmware dose not USB. Bit3-7：Reserved
Byte19-20	XXXX	Local device 32kHz oscillator drift (ppm). Initial setting = 0x0050 (80ppm) This value is decided by each vender. This value influence sync window length and sleep time.
Byte21-22	XXXX	Local device 32kHz jitter drift (us). Initial setting = 0x000A (10us) This value is decided by each vender. This value influence sync window length and sleep time.
Byte25-24	XXXX	Margin to wake up before SniffAttempt During Sniff mode, this value is used to wake up before SniffAttempt. Unit is ms. For example, 0x000a, TC35661 wakes up 10ms before SniffAttmt Host CPU can send sending data during this period..
Byte25-30	000000000000	Reserved. 0x000000000000 fixed.

M2_Deep_Sleep_Set event

Parameters	Value	Parameter Description
Byte0	FF	Event code
Byte1	0A	Command length
Byte2	08	OCF
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	000000	Reserved
Byte8	14	1： Accepter 4： Initiator
Byte9	68	Information ID
Byte10	00	Command result 00：M2MSG_OK(Success) 02：M2MSG_UNKNOWN_DATA_TYPE (information data type is not 18-byte string) 04：M2MSG_INVALID_DATA_VALUE (Not set in [Deep-sleep instructions / Set Notify specific interface])
Byte11	00	Data type 00：No information data

3.4.4 Example for to enable I2C with M2 SET command

This command is used to enable I2C.
This command is need to send after M2_BTL_E2PROM_WRITE_ENABLE command.

M2_BTL_SET_I2C_ENABLE command

Parameters	Value	Parameter Description
Byte0	08	OCF
Byte1	FC	OGF＋OCF
Byte2	0B	Command length
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	000000	Reserved
Byte8	14	1： Accepter 4： Initiator
Byte9	5B	Information ID = BTL_SET_I2C_ENABLE
Byte10	FF	Reserved. 0xFF fixed.
Byte11	02	Data type 02：uint16
Byte12	XX	SCL frequency setting Recommendation is 0x03. Detail information is TBD.
Byte13	XX	Set spike removal filter number. Recommendation is 0x01.Detail information is TBD.

M2_BTL_SET_I2C_ENABLE event

Parameters	Value	Parameter Description
Byte0	FF	Event code
Byte1	0A	Command length
Byte2	08	OCF
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	000000	Reserved
Byte8	14	1： Accepter 4： Initiator
Byte9	5B	Information ID = BTL_SET_I2C_ENABLE
Byte10	00	Command result 00：M2MSG_OK(Success) 02：M2MSG_UNKNOWN_DATA_TYPE (information data type is not 18-byte string) 04：M2MSG_INVALID_DATA_VALUE (Not set in [Deep-sleep instructions / Set Notify specific interface])
Byte11	00	Data type 00：No information data

3.4.5 Example for Host can control GPIO with M2 SET command

To control GPIO by Host CPU. Execution order is as follows.
HCI_UART_RTSCTS_Control -> this command -> M2_GPIO_CONTROL_OUTPUT command

M2_GPIO_CONTROL_ENABLE command

Parameters	Value	Parameter Description
Byte0	08	OCF
Byte1	FC	OGF＋OCF
Byte2	0B	Command length
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	000000	Reserved
Byte8	14	1： Accepter 4： Initiator
Byte9	29	Information ID = BTL_GPIO_CONTROL
Byte10	FF	Reserved (0xFF fixed)
Byte11	02	Data type 02：uint16
Byte12	XX	00: this function is disabled. 01: Host can control GPIO
Byte13	00	Reserved

M2_GPIO_CONTROL_ENABLE event

Parameters	Value	Parameter Description
Byte0	FF	Event code
Byte1	0A	Command length
Byte2	08	OCF
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	000000	Reserved
Byte8	14	1： Accepter 4： Initiator
Byte9	29	Information ID = BTL_GPIO_FREE_CONTROL
Byte10	00	Command result 00：M2MSG_OK(Success) 02：M2MSG_UNKNOWN_DATA_TYPE (Information data type is not Uint16) 04：M2MSG_INVALID_DATA_VALUE FE：UNSPECIFIED_ERROR (Set enable when already set enable. Set disable when already set disable.)
Byte11	00	Data type 00：No information data

3.4.6 Example for GPIO output with M2 SET command

The GPIO_OUTPUT command is used for GPIO output.

< M2_GPIO_CONTROL_ENABLE > command is needed before this command execution.

M2_GPIO_CONTROL_OUTPUT command

Parameters	Value	Parameter Description
Byte0	08	OCF
Byte1	FC	OGF＋OCF
Byte2	0B	Command length
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	000000	Reserved
Byte8	14	1：Accepter 4：Initiator
Byte9	2A	Information ID = BTL_GPIO_OUTPUT
Byte10	FF	Reserved (0xFF fixed)
Byte11	02	Data type 02：uint16
Byte12	XX	Set output level. 00: ”L” level output. 01: “H” level output.
Byte13	XX	Set GPIO number. Example when GPIO10, value is 0x0A

M2_GPIO_CONTROL_OUTPUT event

Parameters	Value	Parameter Description
Byte0	FF	Event code
Byte1	0A	Command length
Byte2	08	OCF
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	000000	Reserved
Byte8	14	1：Accepter 4：Initiator
Byte9	2A	Information ID = BTL_GPIO_OUTPUT
Byte10	00	Command result 00：M2MSG_OK(Success) 02：M2MSG_UNKNOWN_DATA_TYPE (Information data type is not Uint16) 04：M2MSG_INVALID_DATA_VALUE FE：UNSPECIFIED_ERROR (GPIO free control disable)
Byte11	00	Data type 00：No information data

3.4.7 Example for EEPROM write enable with M2 SET command

This command is used for EEPROM write protection enable.
This command is need to send before M2_BTL_SET_I2C_ENABLE.

M2_BTL_E2PROM_WRITE_ENABLE command

Parameters	Value	Parameter Description
Byte0	08	OCF
Byte1	FC	OGF＋OCF
Byte2	09	Command length
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	000000	Reserved
Byte8	14	1： Accepter 4： Initiator
Byte9	83	Information ID = BTL_E2PROM_WRITE_PROTECTION_ENABLE
Byte10	FF	Reserved (0xFF fixed)
Byte11	00	Data type 00：No data

M2_BTL_E2PROM_WRITE_ENABLE event

Parameters	Value	Parameter Description
Byte0	FF	Event code
Byte1	0A	Command length
Byte2	08	OCF
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	000000	Reserved
Byte8	14	1： Accepter 4： Initiator
Byte9	83	Information ID = BTL_E2PROM_WRITE_PROTECTION_ENABLE
Byte10	00	Command result 00：M2MSG_OK(Success) 02：M2MSG_UNKNOWN_DATA_TYPE 04：M2MSG_INVALID_DATA_VALUE
Byte11	00	Data type 00：No information data

3.4.8 Example for I2C-EEPROM data write with M2 SET command

To write the specified data to the specified address in the EEPROM with an I2C interface.
Refer to [5.I2C-EEPROM Setup Steps] to write BD_ADDR.

M2_GENERAL_WRITE_EEPROM command

Parameters	Value	Parameter Description
Byte0	08	OCF
Byte1	FC	OGF＋OCF
Byte2	10-8F	Command length
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	000000	Reserved
Byte8	14	1:Accepter 4: Initiator
Byte9	89	Information ID
Byte10	FF	Reserved (0xFF fixed)
Byte11	10	Data type 10:Byte array (First byte is length)
Byte12	07-86	To specify the parameter length from Byte13. Not including parameter information.
Byte13	XX	To specify the EEPROM device address (8bit). Device Code(4bit)+Slave Address(3bit)+R/W(1bit) Device code: usually 1010. Slave address: it depends on EEPROM setting. When GND, set to 0x00. R/W: Don’t care.
Byte14	XX	To specify the maximum bytes’ number in 1 write cycle. 0x00:8bytes 0x01:16bytes
Byte15	XX	To specify the address bit length. 0x00:8bit addressing 0x01:16bit addressing
Byte16	XX	To specify the written data size The range if from 0x01 to 0x80.
Byte17	XX	To specify the least significant digit (LSD) of the written address. e.g.: When writing to 0x0002, to specify 0x02.
Byte18	XX	To specify the LSD+1 of the written address. e.g.: When writing to 0x0002, to specify 0x00.
Byte19-	XX….	To specify the written data in bytes’ terms. Maximum 0x80bytes.

M2_GENERAL_WRITE_EEPROM event

Parameters	Value	Parameter Description
Byte0	FF	Event code
Byte1	0A	Command length
Byte2	08	OCF
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	000000	Reserved
Byte8	14	1: Accepter 4: Initiator
Byte9	89	Information ID
Byte10	00	Command result 00:successful
Byte11	00	Data type 00:No information data

3.4.9 Example for patch information data (Former data of SWAP) write with M2 SET command

This command is used for patch information data (Former data of SWAP) write.

M2_BTL_PATCH_SWAP_BASE command

Parameters	Value	Parameter Description
Byte0	08	OCF
Byte1	FC	OGF＋OCF
Byte2	2F	Command length
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	000000	Reserved
Byte8	14	1： Accepter 4： Initiator
Byte9	55	Information ID = M2_BTL_PATCH_SWAP_BASE
Byte10	FF	Reserved (0xFF fixed)
Byte11	10	Data type 10：Byte array(First byte is length)
Byte12	25	Parameter length. 0x25 fixed.
Byte13 -	XXXX・・・	Patch information data

M2_BTL_PATCH_SWAP_BASE event

Parameters	Value	Parameter Description
Byte0	FF	Event code
Byte1	0A	Command length
Byte2	08	OCF
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	000000	Reserved
Byte8	14	1： Accepter 4： Initiator
Byte9	55	Information ID = M2_BTL_PATCH_SWAP_BASE
Byte10	00	Command result 00：M2MSG_OK(Success) 02：M2MSG_UNKNOWN_DATA_TYPE 04：M2MSG_INVALID_DATA_VALUE
Byte11	00	Data type 00：No information data

3.4.10 Example for patch program data write with M2 SET command

This command is used for patch program data write.

M2_BTL_PATCH_SWAP_PROG_WRITE command

Parameters	Value	Parameter Description
Byte0	08	OCF
Byte1	FC	OGF＋OCF
Byte2	XX	Command length
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	000000	Reserved
Byte8	14	1：Accepter 4：Initiator
Byte9	56	Information ID = M2_BTL_PATCH_SWAP_PROG_WRITE
Byte10	FF	Reserved (0xFF fixed)
Byte11	10	Data type 10：Byte array(First byte is length)
Byte12	XX	Parameter length.
Byte13 -	XXXX・・・	Patch program data

M2_BTL_PATCH_SWAP_PROG_WRITE event

Parameters	Value	Parameter Description
Byte0	FF	Event code
Byte1	0A	Command length
Byte2	08	OCF
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	000000	Reserved
Byte8	14	1：Accepter 4：Initiator
Byte9	56	Information ID = M2_BTL_PATCH_SWAP_PROG_WRITE
Byte10	00	Command result 00：M2MSG_OK(Success) 02：M2MSG_UNKNOWN_DATA_TYPE 04：M2MSG_INVALID_DATA_VALUE
Byte11	00	Data type 00：No information data

3.4.11 Example for patch control(enable/disable) with M2 SET command

This command is used for patch control(enable/disable).

M2_BTL_PATCH_CONTROL command

Parameters	Value	Parameter Description
Byte0	08	OCF
Byte1	FC	OGF＋OCF
Byte2	0B	Command length
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	000000	Reserved
Byte8	14	1： Accepter 4： Initiator
Byte9	57	Information ID = M2_BTL_PATCH_CONTROL
Byte10	FF	Reserved (0xFF fixed)
Byte11	02	Data type 02：uint16
Byte12-13	XX	Patch control data.

M2_BTL_PATCH_CONTROL event

Parameters	Value	Parameter Description
Byte0	FF	Event code
Byte1	0A	Command length
Byte2	08	OCF
Byte3	00	Reserved
Byte4	A0	Information setting request command
Byte5-7	000000	Reserved
Byte8	14	1： Accepter 4： Initiator
Byte9	57	Information ID = M2_BTL_PATCH_CONTROL
Byte10	00	Command result 00：M2MSG_OK(Success) 02：M2MSG_UNKNOWN_DATA_TYPE 04：M2MSG_INVALID_DATA_VALUE
Byte11	00	Data type 00：No information data

3.4.12 < HCI_M2_Message_Get >

This command is used to get HW and Firmware setting or Module control.

M2Message_Get command

Parameters	Value	Parameter Description
Byte0	08	OCF
Byte1	FC	OGF＋OCF
Byte2	XX	Command length
Byte3	00	Reserved
Byte4	A1	Information getting request command
Byte5-7	00	Reserved
Byte8	X4	X： Accepter
Byte9	XX	Information ID
Byte10	FF	Reserved (0xFF fixed)
Byte11	XX	Data type 00：No information data 01：uint8 02：uint16 03：uint32 04：uint64 81：int8 82：int16 83：int32 84：int64 0F：Strings (0x00 terminated) 10：Byte array(First byte is length)
Byte12-		Data (If Data type is 0x00, Data doesn't exist.)

M2Message_Get event

Parameters	Value	Parameter Description
Byte0	FF	Event code
Byte1	XX	Command length
Byte2	08	OCF
Byte3	00	Reserved
Byte4	A1	Information getting request command
Byte5-7	00	Reserved
Byte8	X4	X：Accepter
Byte9	XX	Information ID
Byte10	XX	Command result 00：successful 01：Not supported Information ID 02：Not supported Information type 03：Data size exceed byte length when byte array transmit 04：Information data error 05：Voice processing executing 06：Audio processing executing 07：Voice recognition processing executing 08：Resource lack FE：Other errors Others：Error
Byte11	XX	Data type 00：No information data 01：uint8 02：uint16 03：uint32 04：uint64 81：int8 82：int16 83：int32 84：int64 0F：Strings (0x00 terminated) 10：Byte array(First byte is length)
Byte12-		Data (If Data type is 0x00，Data doesn't exist.)

Next Page show example for this command

3.4.13 Example for firmware version with M2 GET command

Getting Firmware Version command

Parameters	Value	Parameter Description
Byte0	08	OCF
Byte1	FC	OGF＋OCF
Byte2	09	Command length
Byte3	00	Reserved
Byte4	A1	Information getting request command
Byte5-7	00	Reserved
Byte8	14	1：Accepter 4：Initiator
Byte9	0D	Information ID=Get firmware version
Byte10	FF	Reserved (0xFF fixed)
Byte11	00	Data type 00：No information data

Getting Firmware Version event

Parameters	Value	Parameter Description
Byte0	FF	Event code
Byte1	16	Command length
Byte2	08	OCF
Byte3	00	Reserved
Byte4	A1	Information getting request command
Byte5-7	00	Reserved
Byte8	14	1：Accepter
Byte9	0D	Information ID=Get firmware version
Byte10	00	Command result 00：successful
Byte11	0F	Data type 0F：Strings (0x00 terminated)
Byte12-23		Data Version Format A.BB.CCD-EE A: Product: 9(indicate TC35661). BB: firm main version1 CC: firm main version2 D: Firm mariner version1 EE: Firm mariner version2 For example: (Byte12) 39 2e 30 30 2e 36 32 50 2d 30 37 00 (Byte23) 9.00.62P-07 Last data should be 0x00(Null).

3.4.14 Example for I2C-EEPROM data read with M2 GET command

To read the specified data from the specified address in the EEPROM with an I2C interface.
M2_GENERAL_READ_EEPROM event is response.
Refer to [5.I2C-EEPROM Setup Steps] to read BD_ADDR.

M2_GENERAL_READ_EEPROM command

Parameters	Value	Parameter Description
Byte0	08	OCF
Byte1	FC	OGF＋OCF
Byte2	0D	Command length
Byte3	00	Reserved
Byte4	A1	Information getting request command
Byte5-7	000000	Reserved
Byte8	14	9: Accepter 4: Initiator
Byte9	88	Information ID
Byte10	FF	Reserved (0xFF fixed)
Byte11	10	Data type 10:Byte array(First byte is length)
Byte12	04,06	To specify the parameter length from Byte13. Not including parameter information.
Byte13	XX	To specify the EEPROM device address (8bit). Including Device Code(4bit)+Slave Address(3bit)+R/W(1bit) Device code: usually 1010. Slave address: it depends on EEPROM setting. When GND, set to 0x00. R/W: Don't care.
Byte14	XX	To specify the bit length of address. 0x00:8bit addressing 0x01:16bit addressing
Byte15	XX	To specify the reading type. 0x00: current read 0x01: random read To specify the read address by byte17, 18 only in case of random read.
Byte16	XX	To specify the read data size The range if from 0x01 to 0x80.
Byte17	XX	To specify the least significant digit (LSD) of the read address. e.g.: When reading from 0x0002, to specify 0x02. Make sure to specify when byte15 is 0x01(random read).
Byte18	XX	To specify the LSD+1 of the read address. e.g.: When reading from 0x0002, to specify 0x00. Make sure to specify when byte15 is 0x01(random read).

M2_GENERAL_READ_EEPROM event

Parameters	Value	Parameter Description
Byte0	FF	Event code
Byte1	0B-8A	Command length
Byte2	08	OCF
Byte3	00	Reserved
Byte4	A1	Information getting request command
Byte5-7	000000	Reserved
Byte8	14	1: Accepter 4: Initiator
Byte9	88	Information ID
Byte10	00	Command result 00:successful
Byte11	10	Data type 10:Byte array(First byte is length)
Byte12	01-80	To specify the parameter length from Byte13. Not including parameter information.
Byte13	XX….	Read data. Maximum 0x80 bytes.

3.4.15 Example for read the GPIO status with M2 GET command

To read the input level of specified GPIO status (H level or L level).
< M2_GPIO_CONTROL_ENABLE > command is needed before this command execution.

M2_GET_READ_GPIO_STATUS command

Parameters	Value	Parameter Description
Byte0	08	OCF
Byte1	FC	OGF＋OCF
Byte2	0B	Command length
Byte3	00	Reserved
Byte4	A1	Information getting request command
Byte5-7	000000	Reserved
Byte8	14	1：Accepter / 4：Initiator
Byte9	2A	Information ID = BTL_GPIO_READ
Byte10	FF	Reserved (0xFF fixed)
Byte11	02	Data type 02：uint16
Byte12	XX	Set expect level. 00: ”L” level. 01: “H” level.
Byte13	XX	Set GPIO number.

M2_GET_READ_GPIO_STATUS　event

Parameters	Value	Parameter Description
Byte0	FF	Event code
Byte1	0B	Command length
Byte2	08	OCF
Byte3	00	Reserved
Byte4	A1	Information getting request command
Byte5-7	000000	Reserved
Byte8	14	1：Accepter / 4：Initiator
Byte9	2A	Information ID = BTL_GPIO_OUTPUT
Byte10	00	Command result 00：M2MSG_OK(Success) 02：M2MSG_UNKNOWN_DATA_TYPE (Information data type is not Uint16) 04：M2MSG_INVALID_DATA_VALUE FE：UNSPECIFIED_ERROR(GPIO free control disable)
Byte11	01	Data type 01：uint8
Byte12	XX	Information data. 00：Input level is not expected value. 01：Input level is expected value. (Note) This value is not just input data. This is comparison result.

4. Bluetooth Test command

This chapter explains a Bluetooth qualification test mode.

4.1 Sequence

UART log is as follows.
CMD:HOST->TC35661 EVT:TC35661->HOST

```
(1)Write_Scan_Enable- Op=0x0c1a length=0x01 Enable=0x02(Page Scan)
(CMD) 01 1a 0c 01 02
(EVT) 04 0e 04 04 1a 0c 00
Command_Complete- NHCP=0x04 Op=0x0c1a(Write_Scan_Enable) Status=0x00(SUCCESS)

(2)Set_Event_Filter- Op=0x0c05 length=0x03 Type=0x02(Conn-Setup) Condition=0x00(All Devices)
Auto_Acc=0x02(on)
(CMD) 01 05 0c 03 02 00 02
(EVT) 04 0e 04 04 05 0c 00
Command_Complete- NHCP=0x04 Op=0x0c05(Set_Event_Filter) Status=0x00(SUCCESS)

(3)Enable_Device_Under_Test_Mode- Op=0x1803 length=0x00
(CMD) 01 03 18 00
(EVT) 04 0e 04 04 03 18 00
Command_Complete- NHCP=0x04 Op=0x1803(Enable_Device_Under_Test_Mode) Status=0x00(SUCCESS)
```

4.2 Bluetooth test command

4.2.1 < HCI_Write_Scan_Enable >

This command writes the value for the Scan Enable configuration parameter.

HCI_WRITE_SCAN_ENABLE

	Setting value	content
Byte 0	1A	OCF
Byte 1	0C	OGF+OCF
Byte 2	01	Parameter Value length
Byte 3	02	Page Scan enable

HCI_WRITE_SCAN_ENABLE Event

	Setting value	content
Byte 0	0E	Event code
Byte 1	04	Command length
Byte 2	04	Num_HCI_Command_Packets
Byte 3	1A	Commnad_Opcode
Byte 4	0C	Commnad_Opcode
Byte 5	00	Status 00: Success Except 00: Fault

4.2.2 < HCI_SET_EVENT_FILTER >

The Set_Event_Filter command is used by the Host to specify different event filters.

When this command is used for Bluetooth qualification test, set as the following.

HCI_SET_EVENT_FILTER

	Setting value	content
Byte 0	05	OCF
Byte 1	0C	OGF+OCF
Byte 2	03	Parameter Value length
Byte 3	02	Type 02: Conn-Setup
Byte 4	00	Condition 00: All Devices
Byte 5	02	Auto Accept for Qualification. 02: On

HCI_SET_EVENT_FILTER Event

	Setting value	content
Byte 0	0E	Event code
Byte 1	04	Command length
Byte 2	04	Num_HCI_Command_Packets
Byte 3	05	Commnad_Opcode
Byte 4	0C	Commnad_Opcode
Byte 5	00	Status 00: Success Except 00: Fault

4.2.3 < HCI_ENABLE_DEVICE_UNDER_TEST_MODE >

The Enable_Device_Under_Test_Mode command allows the local Bluetooth module to enter test mode via LMP test commands.

HCI_ENABLE_DEVICE_UNDER_TEST_MODE

	Setting value	content
Byte 0	03	OCF
Byte 1	18	OGF+OCF
Byte 2	00	Parameter Value length

HCI_ENABLE_DEVICE_UNDER_TEST_MODE Event

	Setting value	content
Byte 0	0E	Event code
Byte 1	04	Command length
Byte 2	04	Num_HCI_Command_Packets
Byte 3	01	Commnad_Opcode
Byte 4	18	Commnad_Opcode
Byte 5	00	Status 00: Success Except 00: Fault

5. I2C-EEPROM Setup Steps

5.1 Examples for writing to I2C-EEPROM (ST Microelectronics M24C32)

(CMD): Host CPU→TC356661 (EVT)TC35661→Host CPU

・I2C Enable(M2_I2C_ENABLE command)

(CMD)01 08 fc 0b 00 a0 00 00 00 14 5b ff 02 03 01
(EVT)04 ff 0a 08 00 a0 00 00 00 14 5b 00 00

・I2C-EEPROM Write Enable(M2_EEPROM_ENABLE command)

(CMD)01 08 fc 09 00 a0 00 00 00 14 83 ff 00
(EVT)04 ff 0a 08 00 a0 00 00 00 14 83 00 00

・I2C-EEPROM　Data Writing Enable(M2_GENERAL_WRITE_EEPROM command)

To specify writing 02041048cafe to address 0x0002-0x0007
(CMD)01 08 fc 16 00 a0 00 00 00 14 89 ff 10 0c a0 00 01 06 02 00 02 04 10 48 ca fe
(EVT)04 ff 0a 08 00 a0 00 00 00 14 89 00 00

5.2 Examples for reading from I2C-EEPROM (ST Microelectronics M24C32)

(CMD): Host CPU→TC356661 (EVT)TC35661→Host CPU

・I2C Enable(M2_I2C_ENABLE command)

(CMD)01 08 fc 0b 00 a0 00 00 00 14 5b ff 02 03 01

(EVT)04 ff 0a 08 00 a0 00 00 00 14 5b fe 00

・I2C-EEPROM Data Readout(M2_GENERAL_READ_EEPROM command)

To specify it as Random Read and read 1byte from address 0x0002. Then 02 will be read out.
(CMD)01 08 fc 10 00 a1 00 00 00 14 88 ff 10 06 a0 01 01 01 02 00
(EVT)04 ff 0c 08 00 a1 00 00 00 14 88 00 10 01 02

・I2C-EEPROM Data Readout(M2_GENERAL_READ_EEPROM command)

To specify it as Current Read and read 5bytes.
Then 041048 will be read out from address 0x0003-0x0007.
(CMD)01 08 fc 0e 00 a1 00 00 00 14 88 ff 10 04 a0 01 00 05
(EVT)04 ff 10 08 00 a1 00 00 00 14 88 00 10 05 04 10 48 ca fe

End of document