

Bluetooth Baseband LSI  
**TC35661-ROM501**

Deep Sleep Function

July 2014

- TOSHIBA 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 TOSHIBA 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 TOSHIBA products could cause loss of human life, bodily injury or damage to property.  
In developing your designs, please ensure that TOSHIBA products are used within specified operating ranges as set forth in the most recent TOSHIBA products specifications.  
Also, please keep in mind the precautions and conditions set forth in the "Handling Guide for Semiconductor Devices," or "TOSHIBA Semiconductor Reliability Handbook" etc..
- The Toshiba 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 Toshiba 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 Toshiba 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 TOSHIBA 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 TOSHIBA 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.

[Revised Note]

Date	Modification
25 <sup>th</sup> -Aug-2014	1st Edition

Contents

<b>1.</b>	<b>Deep Sleep Function .....</b>	<b>5</b>
1.1.	General Description .....	5
1.2.	Command Procedures .....	5
1.2.1.	First Step .....	5
1.2.2.	Second Steps .....	5
1.3.	Conditions to Enter Deep Sleep Mode .....	5
1.4.	Commands in Detail .....	5
1.4.1.	M2_BTL_SET_DEEP_SLEEP .....	5
1.4.2.	GPIO .....	7
1.4.3.	TCU_MNG_DEEP_SLEEP_REQ .....	7
1.5.	Procedure Examples .....	9
1.5.1.	Deep Sleep Mode by GPIO (SPP) .....	9
1.5.2.	Deep Sleep Mode by GPIO (LE) .....	9
1.5.3.	Deep Sleep Mode by UART command (SPP) .....	9
1.5.4.	Deep Sleep Mode by UART command (LE) .....	9
1.6.	Function Examples .....	10
1.6.1.	GPIO Controlled .....	10
1.6.2.	UART Controlled .....	13
1.7.	Message Sequence Chart Examples .....	15
1.7.1.	SPP Operation .....	15
1.7.2.	LE Operation .....	17
1.8.	HostWakeup Signal .....	19
1.8.1.	HostWakeUp Description .....	19
1.8.2.	The Example of Sequence Using HostWakeUp Signal .....	19

## 1. Deep Sleep Function

### 1.1. General Description

Deep sleep function is available for TC35661 to reduce power consumption. TC35661 uses an external 32-kHz clock for deep sleep mode. In order to enter deep sleep mode, TC35661 requires jitter and drift information of the 32-kHz clock. The information is different from that of 26 MHz clock for active mode.

### 1.2. Command Procedures

#### 1.2.1. First Step

During HCI mode, M2\_BT\_L\_SET\_DEEP\_SLEEP command is required to set deep sleep clock parameters such as drift and jitter for 32-kHz clock.

#### 1.2.2. Second Steps

During complete mode, two ways are available for host CPU to enable deep sleep mode for TC35661.

- 1) GPIO0 (RequestWakeUp)
- 2) UART command (TCU\_MNG\_DEEP\_SLEEP\_REQ)

### 1.3. Conditions to Enter Deep Sleep Mode

Deep sleep mode is available for the following statuses:

- No Bluetooth link (SPP and LE)
- Sniff mode (SPP)
- Advertising (LE)
- Connected (LE)

### 1.4. Commands in Detail

#### 1.4.1. M2\_BT\_L\_SET\_DEEP\_SLEEP

M2\_BT\_L\_SET\_DEEP\_SLEEP command sets 32-kHz clock drift and jitter for deep sleep mode. The jitter and drift values are given by the vendor who have supplied the 32-kHz clock crystal. The values are used to determine sync window length for RF receiving. Therefore, when the values are smaller than the given values, sniff link might be disconnected. On the other hand, when the values are bigger than the given values, the sync window for RF receiving opens widely, hence the power consumption is increased.

M2\_BT\_L\_SET\_DEEP\_SLEEP command (For more detail, please refer to "TC35661APL\_ROM501\_Extension HCI\_E\_XXXXXX.pdf")

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	Set the value for Initiator and Acceptor 1: value of Acceptor 4: value of 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	XXXXXXX	Crystal stabilization time (us). Initial setting = 0x0000BB8 (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:value of Acceptor 4:value of 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

**1.4.2. GPIO**

Host CPU controls GPIO0 to enable or disable deep sleep mode. TC35661 notifies its status (active mode or deep sleep mode) with GPIO1. M2\_BTL\_SET\_DEEP\_SLEEP command sets method at Bit0: GPIO (0=No notification / 1=Notification) in Byte18 (deep-sleep instructions/set notify specific interface).

GPIO0 = H: deep sleep mode is not available.

GPIO0 = L: deep sleep mode is available.

GPIO1 = H: deep sleep mode. Host cannot send UART command.

GPIO1 = L: active mode. Host can send UART command.

**1.4.3. TCU\_MNG\_DEEP\_SLEEP\_REQ**

TCU\_MNG\_DEEP\_SLEEP\_REQ command is used in complete mode to enter deep sleep mode or to wake up. After a host has sent TCU\_MNG\_DEEP\_SLEEP\_REQ(Enable) command, the host shall send TCU\_MNG\_DEEP\_SLEEP\_REQ(Disable) command before sending other commands. TC35661 enters deep sleep mode after receiving TCU\_MNG\_DEEP\_SLEEP\_REQ(Enable) command. During deep sleep mode, TC35661 recognizes all commands as TCU\_MNG\_DEEP\_SLEEP\_REQ(Disable) command.

Command Format: (For more detail, please refer to "TC35661APL\_ROM501\_MNG\_E\_xxxxxx.pdf")

ServiceID	1 Byte
OpCode	1 Byte
Parameter_Length	2 Bytes
mode	1 Bytes

ServiceID: 0xE1

OpCode: 0xB6

Parameter Length: 0x0001

Parameters:

Parameters	Parameter Description	Value
mode	Deep Sleep mode setting	
	Disable	0x00
	Enable	0x01



## 1.5. Procedure Examples

### 1.5.1. Deep Sleep Mode by GPIO (SPP)

- 1) Send M2\_BTL\_SET\_DEEP\_SLEEP command during HCI mode with Bit0 = 1 in Byte18 (GPIO notification).
- 2) Send TCU\_MNG\_INIT\_REQ command during complete mode (32-kHz clock is enabled).
- 3) Input high to GPIO0 (No Sleep mode).
- 4) Connect SPP.
- 5) Send TCU\_MNG\_SNIFF\_MODE\_CONTROL\_REQ command to set sniff mode.
- 6) Input Low to GPIO0 to enter deep sleep mode.

### 1.5.2. Deep Sleep Mode by GPIO (LE)

- 1) Send M2\_BTL\_SET\_DEEP\_SLEEP command during HCI mode with Bit0 = 1 in Byte18 (GPIO notification).
- 2) Send TCU\_MNG\_INIT\_REQ command during complete mode (32-kHz clock is enabled).
- 3) Send TCU\_MNG\_LE\_INIT\_REQ command during complete mode.
- 4) Input high to GPIO0 (No Sleep mode).
- 5) Start advertising or get connected.
- 6) Input Low to GPIO0 to enter deep sleep mode.

### 1.5.3. Deep Sleep Mode by UART command (SPP)

- 1) Send M2\_BTL\_SET\_DEEP\_SLEEP command during HCI mode with Bit1 = 1 in Byte18 (UART notification)
- 2) Send TCU\_MNG\_INIT\_REQ command during complete mode (32-kHz clock is enabled).
- 3) Connect SPP.
- 4) Send TCU\_MNG\_SNIFF\_MODE\_CONTROL\_REQ command to set sniff mode.
- 5) Send TCU\_MNG\_DEEP\_SLEEP\_REQ command to enter deep sleep mode.

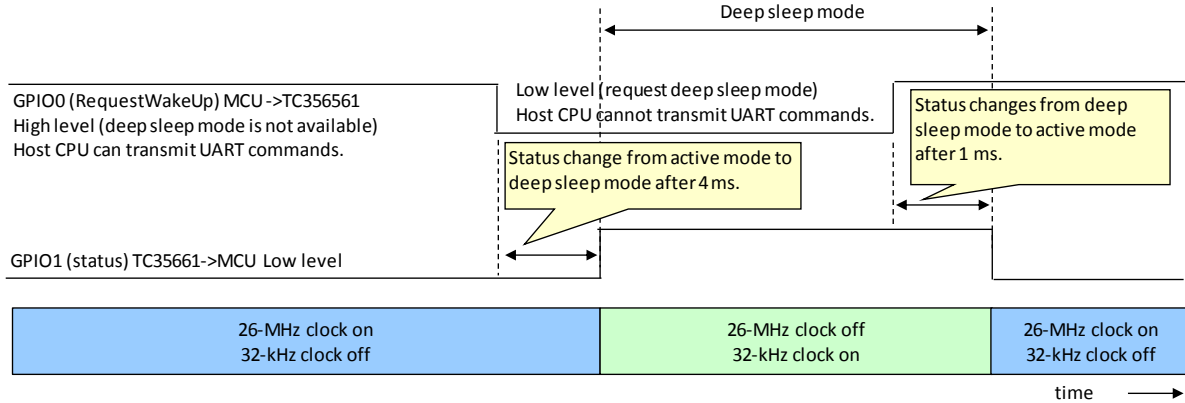
### 1.5.4. Deep Sleep Mode by UART command (LE)

- 1) Send M2\_BTL\_SET\_DEEP\_SLEEP command during HCI mode with Bit1 = 1 in Byte18 (UART notification)
- 2) Send TCU\_MNG\_INIT\_REQ command during complete mode (32-kHz clock is enabled).
- 3) Send TCU\_MNG\_LE\_INIT\_REQ command during complete mode.
- 4) Start advertising or get connected.
- 5) Send TCU\_MNG\_DEEP\_SLEEP\_REQ command to enter deep sleep mode.

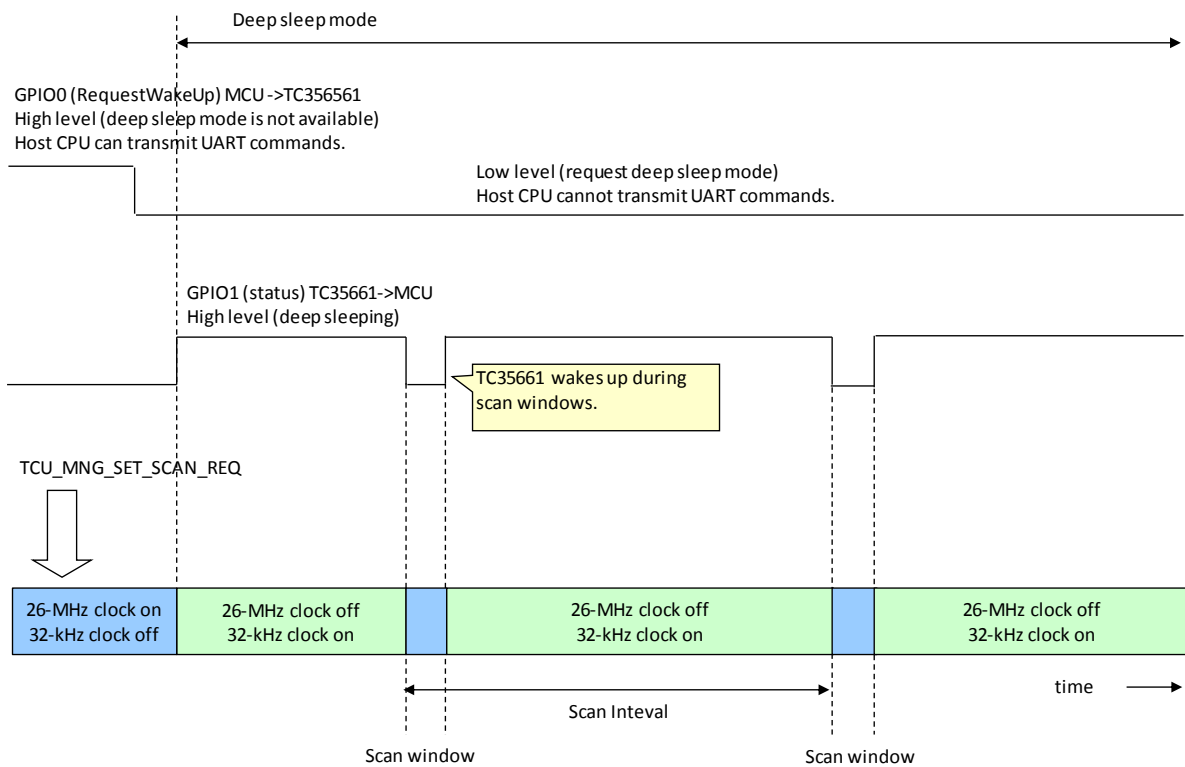
## 1.6. Function Examples

### 1.6.1. GPIO Controlled

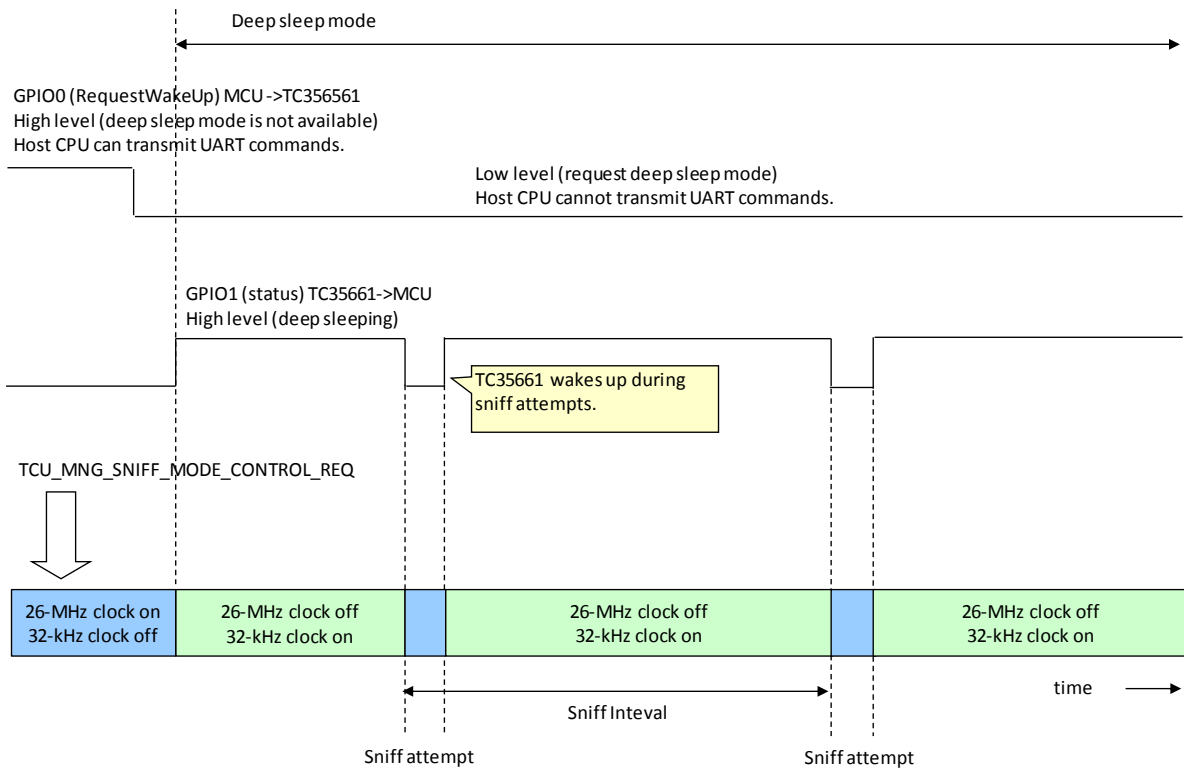
#### During No Bluetooth Link



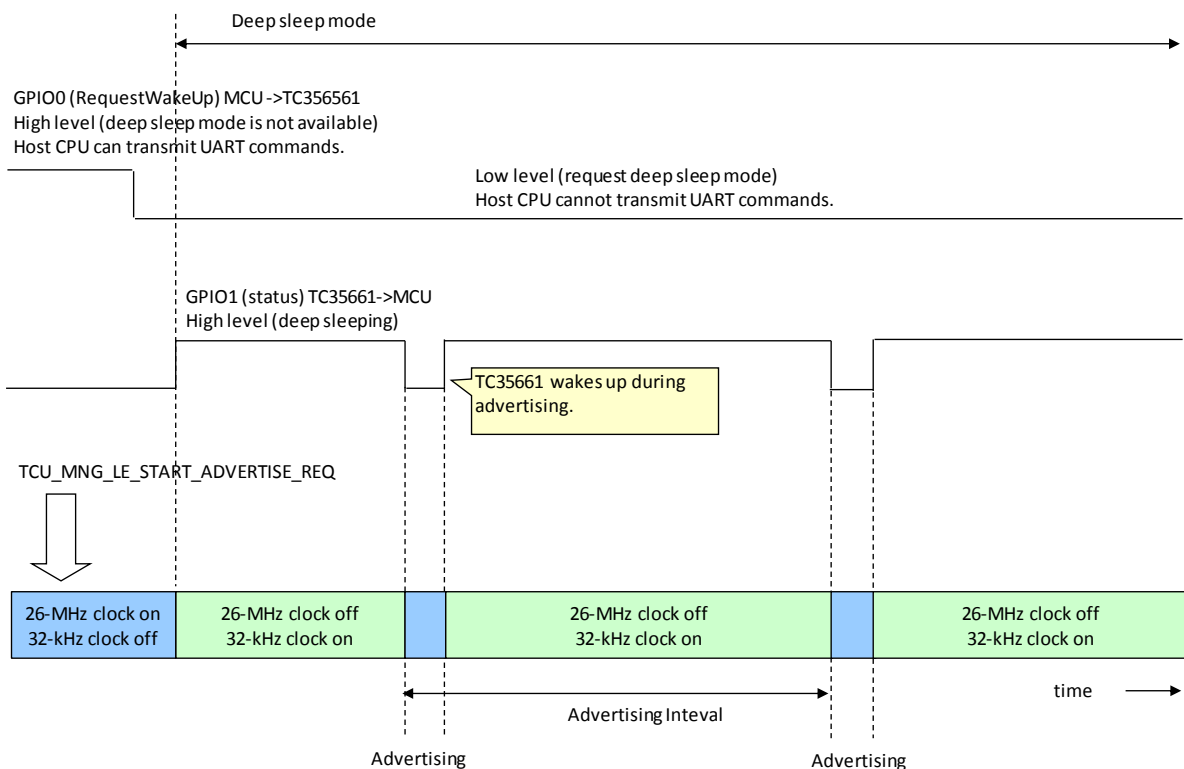
#### During Scan mode



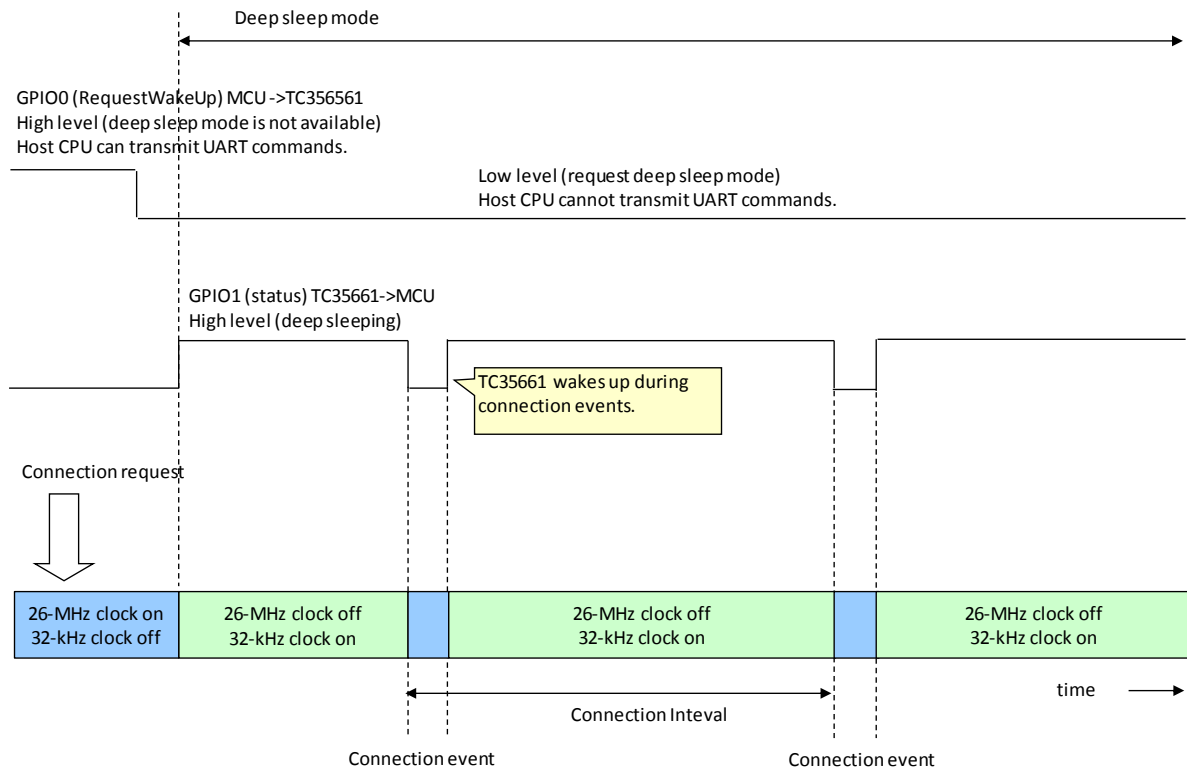
## During Sniff mode



## During Advertising

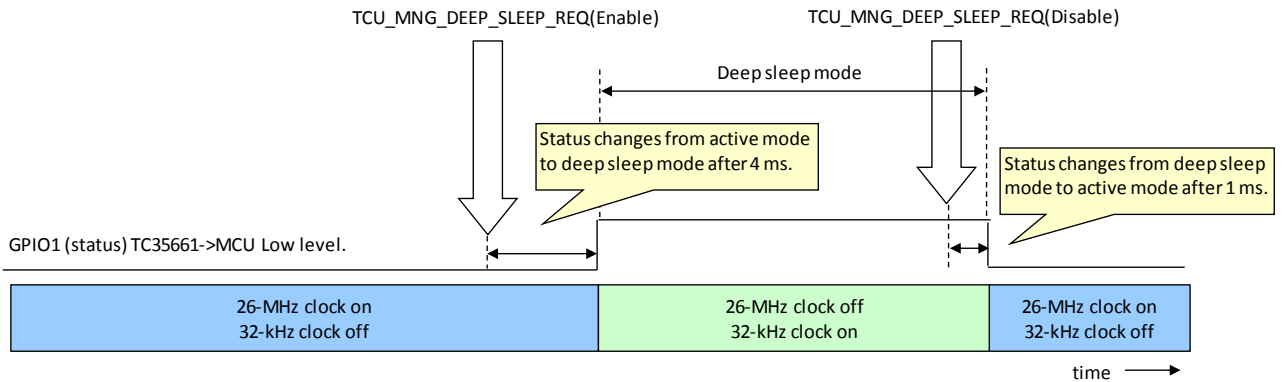


## During LE Connection

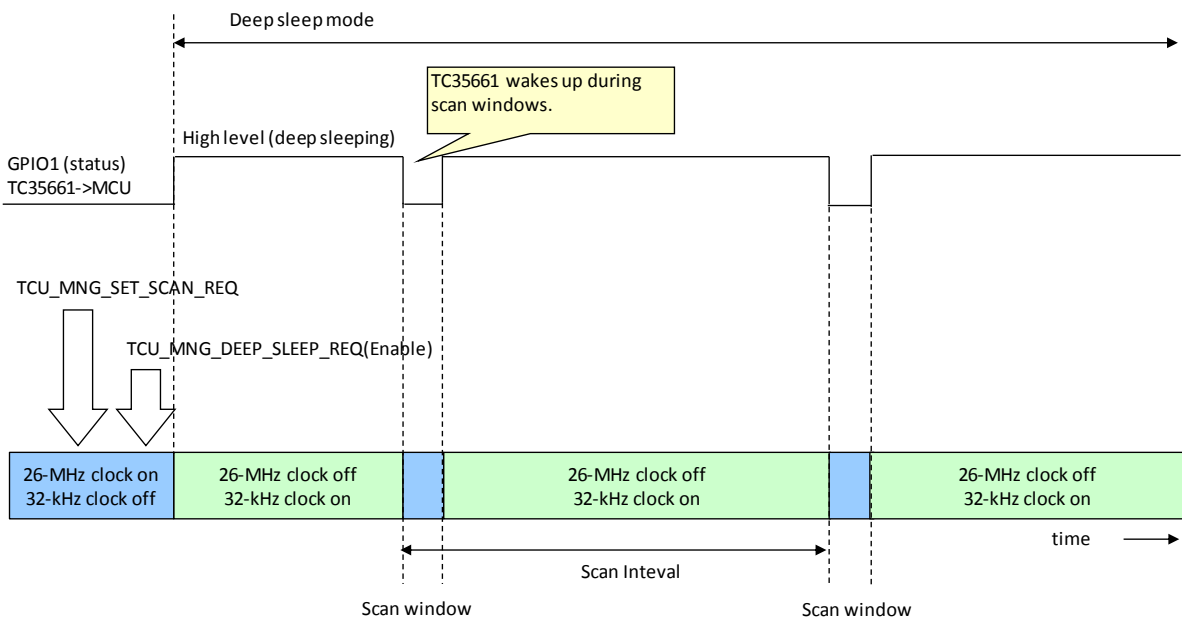


## 1.6.2. UART Controlled

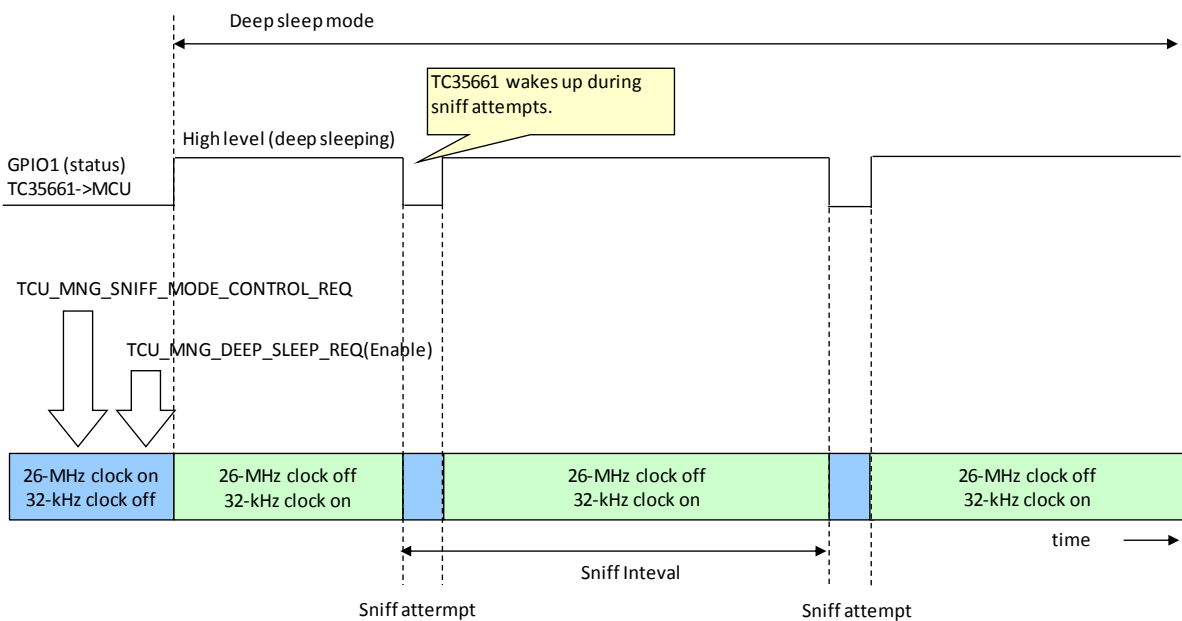
### During No Bluetooth Link



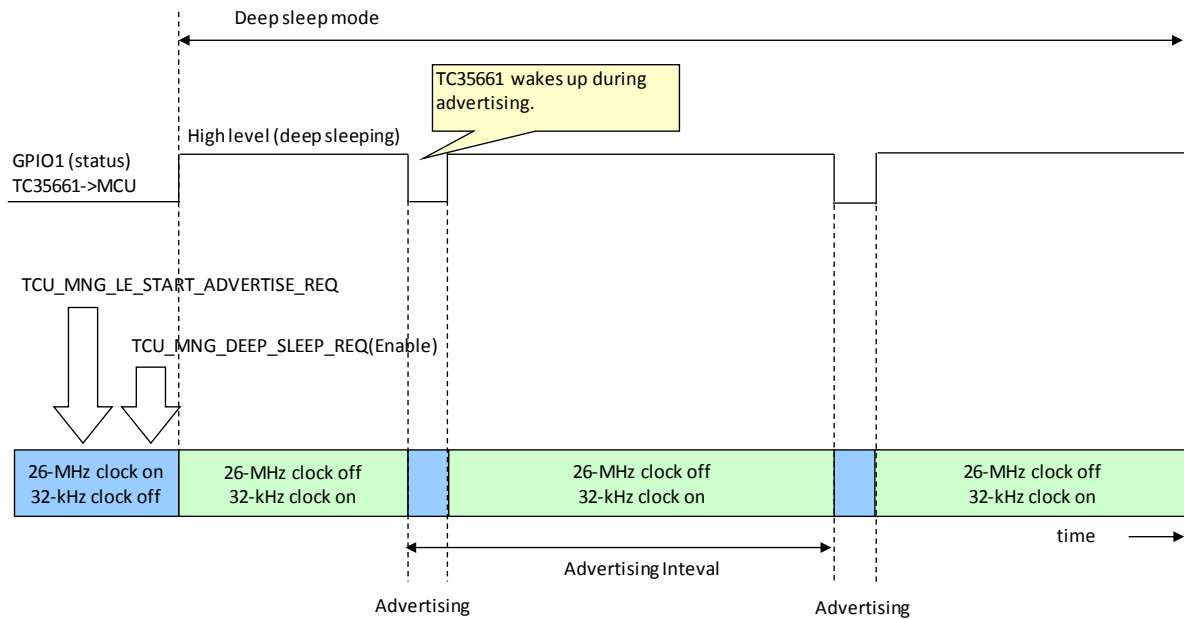
### During Scan Mode



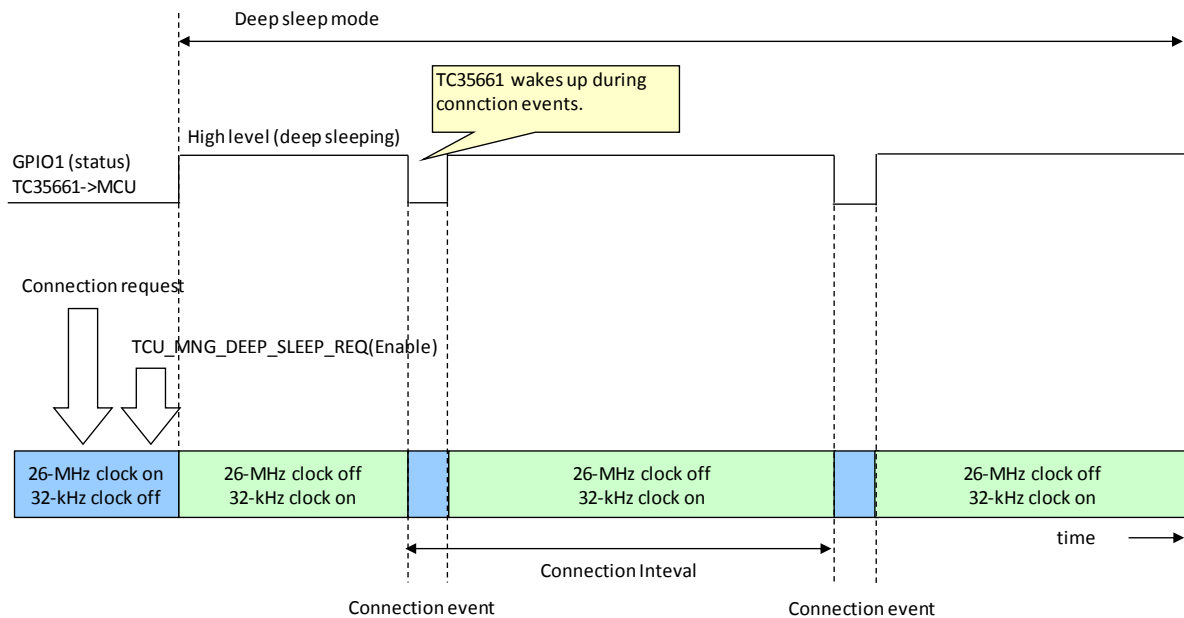
### During Sniff Mode



## During Advertising

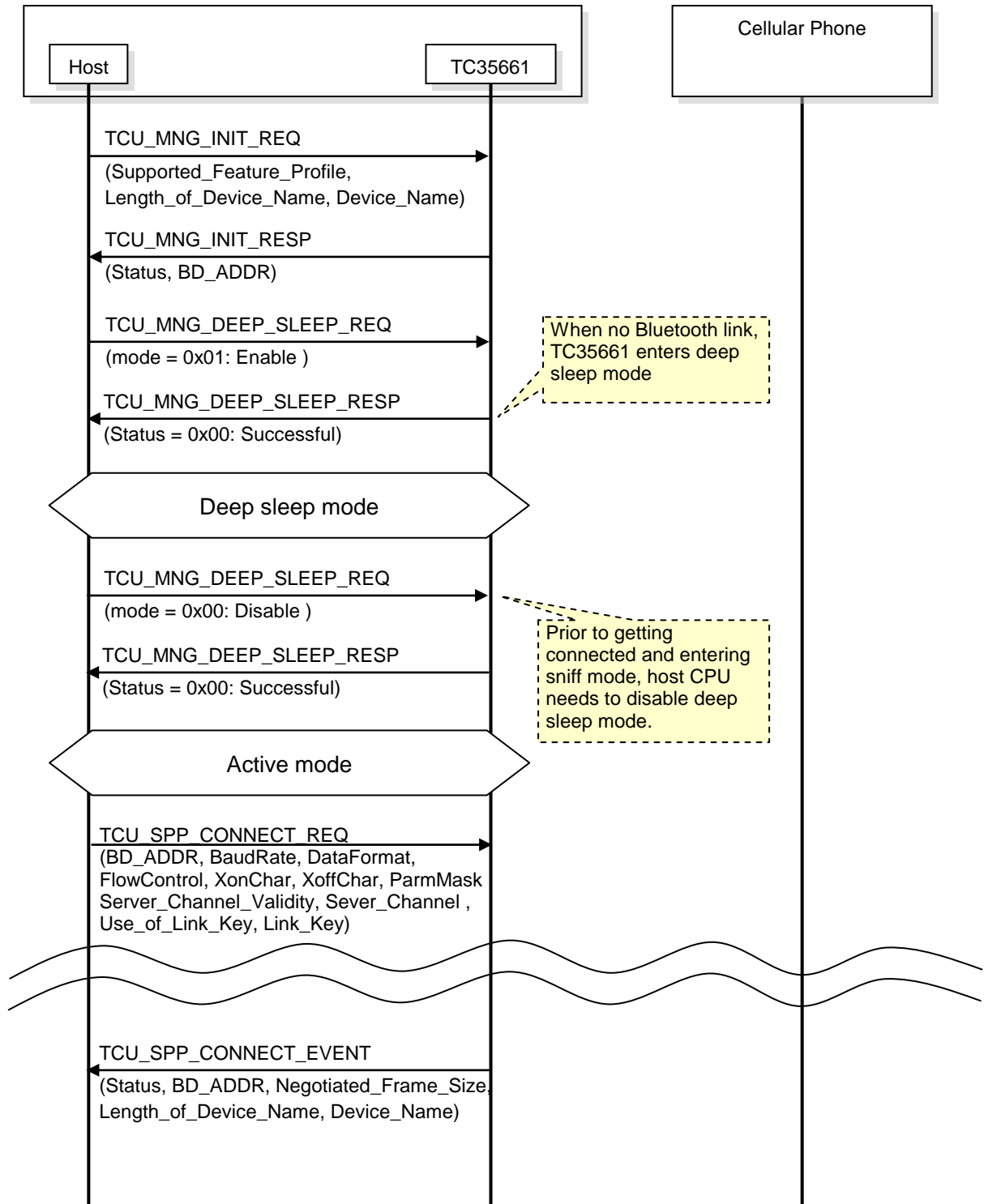


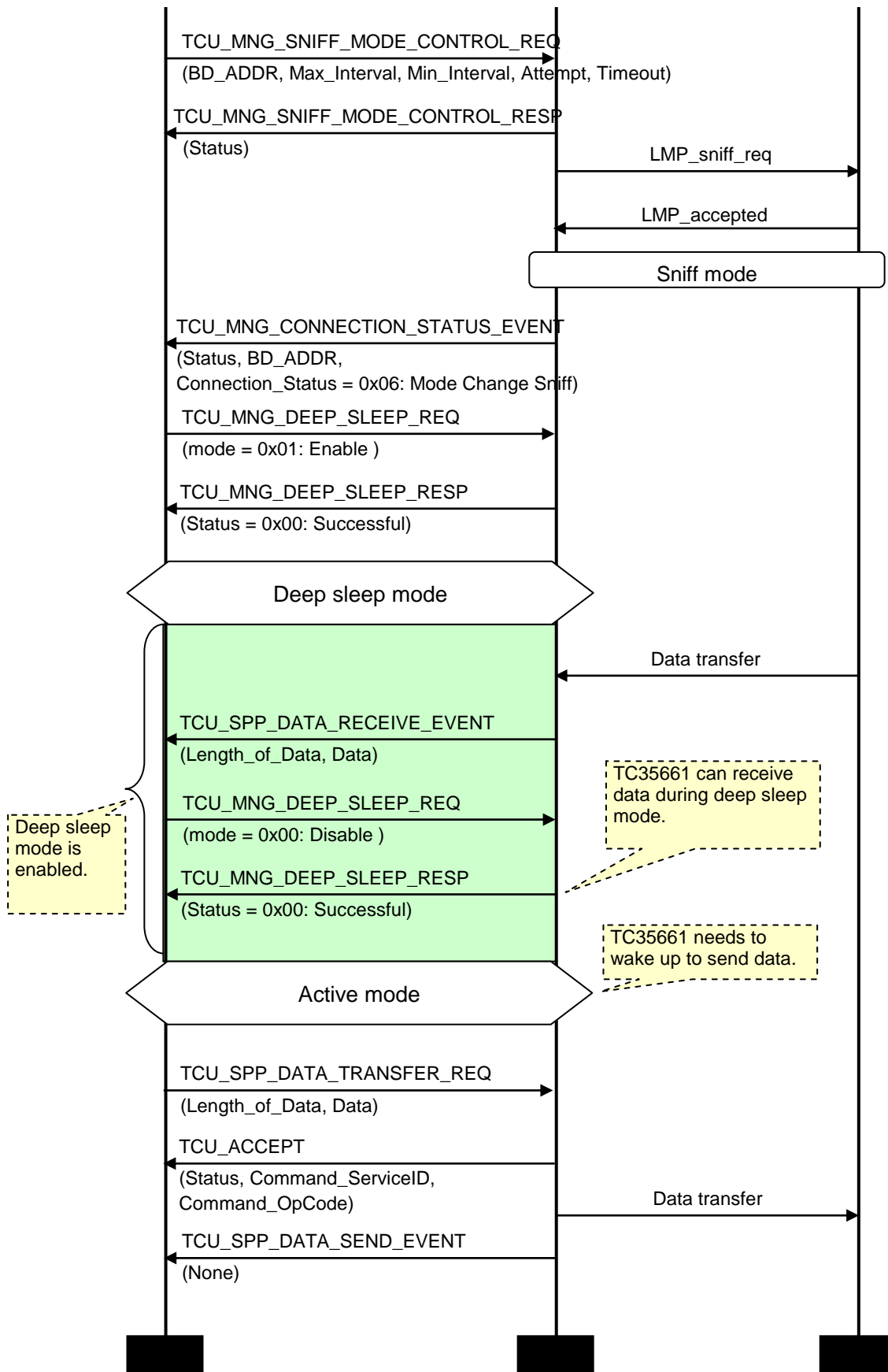
## During LE Connection



## 1.7. Message Sequence Chart Examples

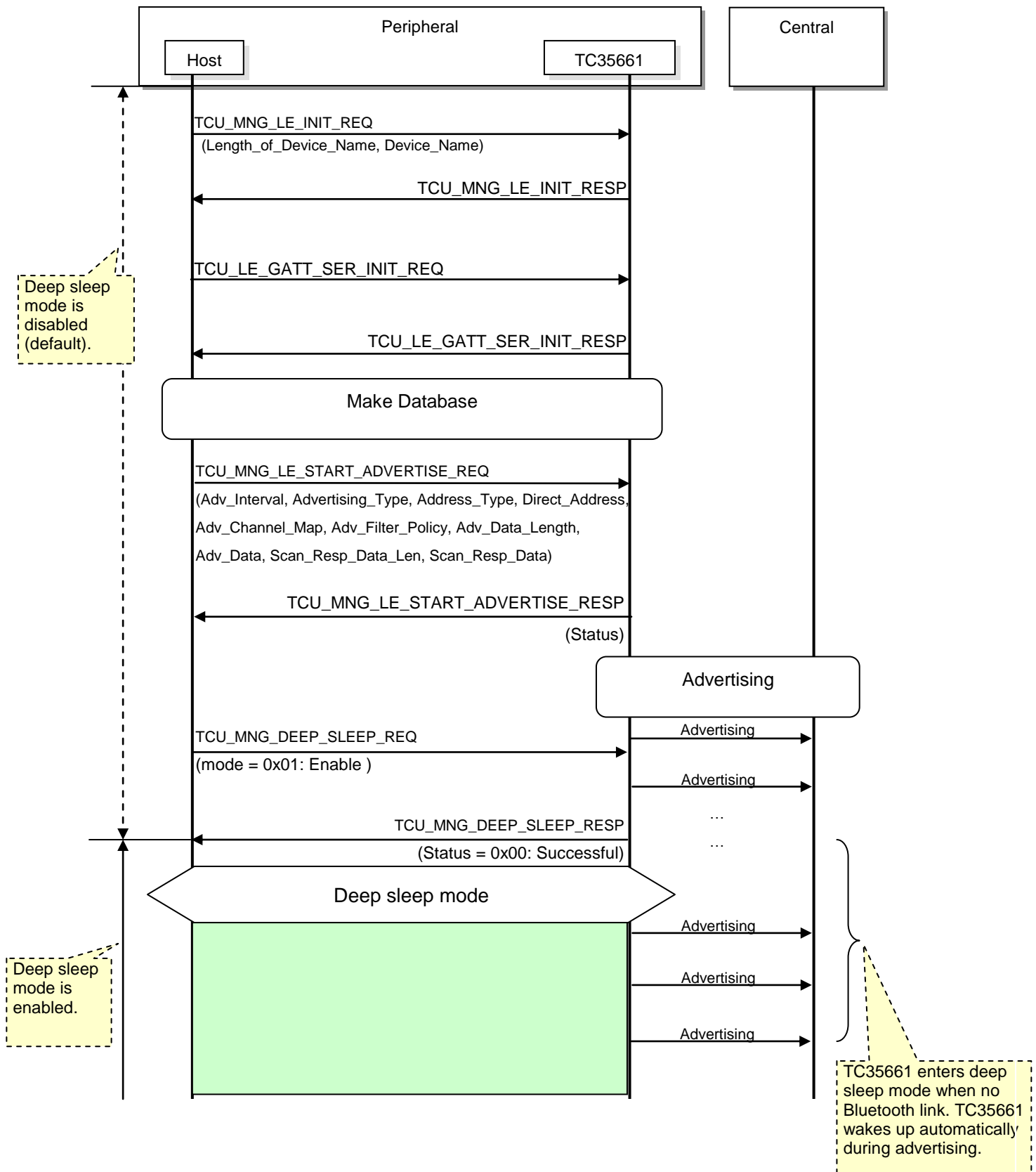
### 1.7.1. SPP Operation

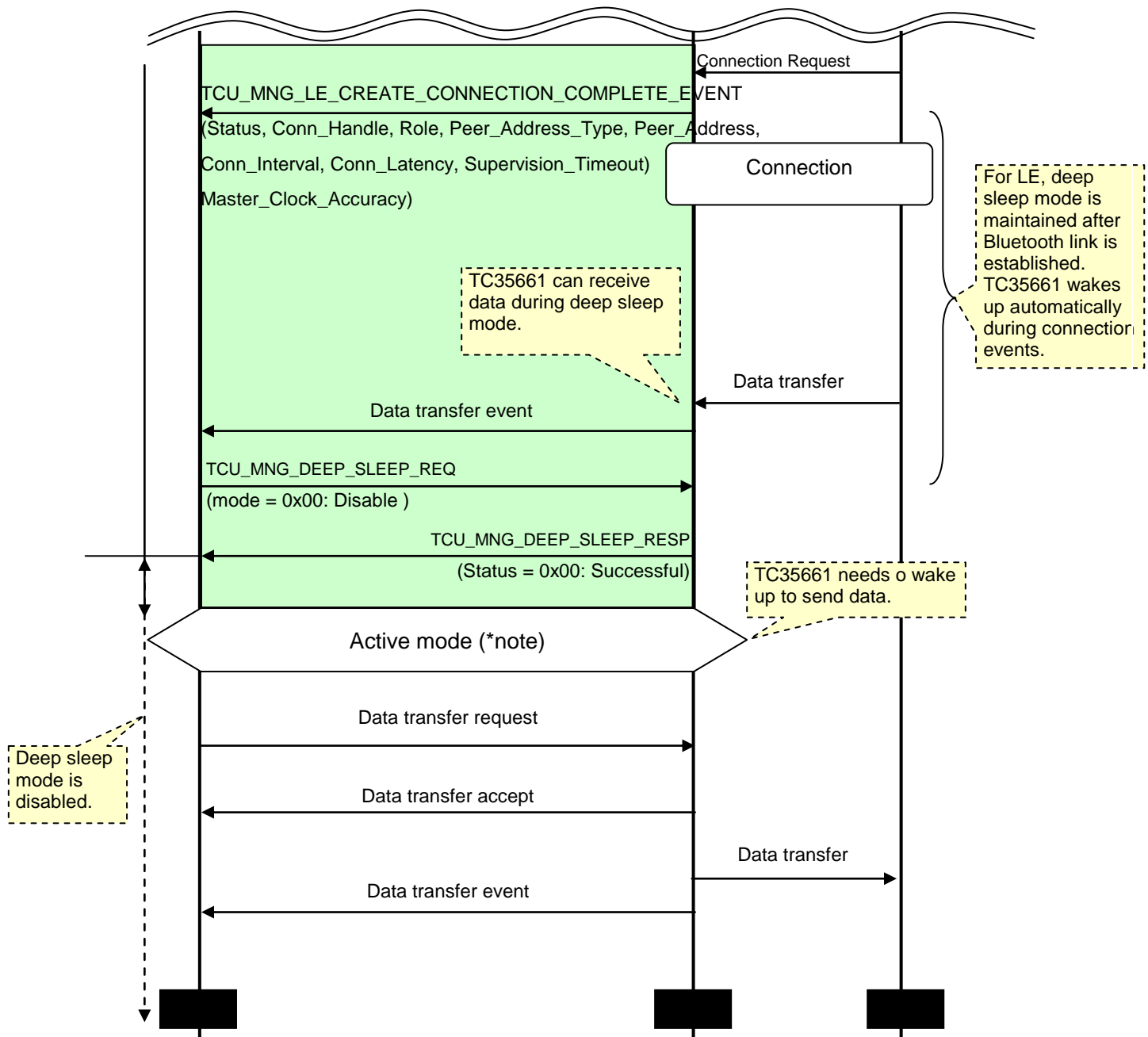






### 1.7.2. LE Operation





**\*Note**

During deep sleep mode, any command from host CPU wakes up TC35661.

## 1.8. HostWakeup Signal

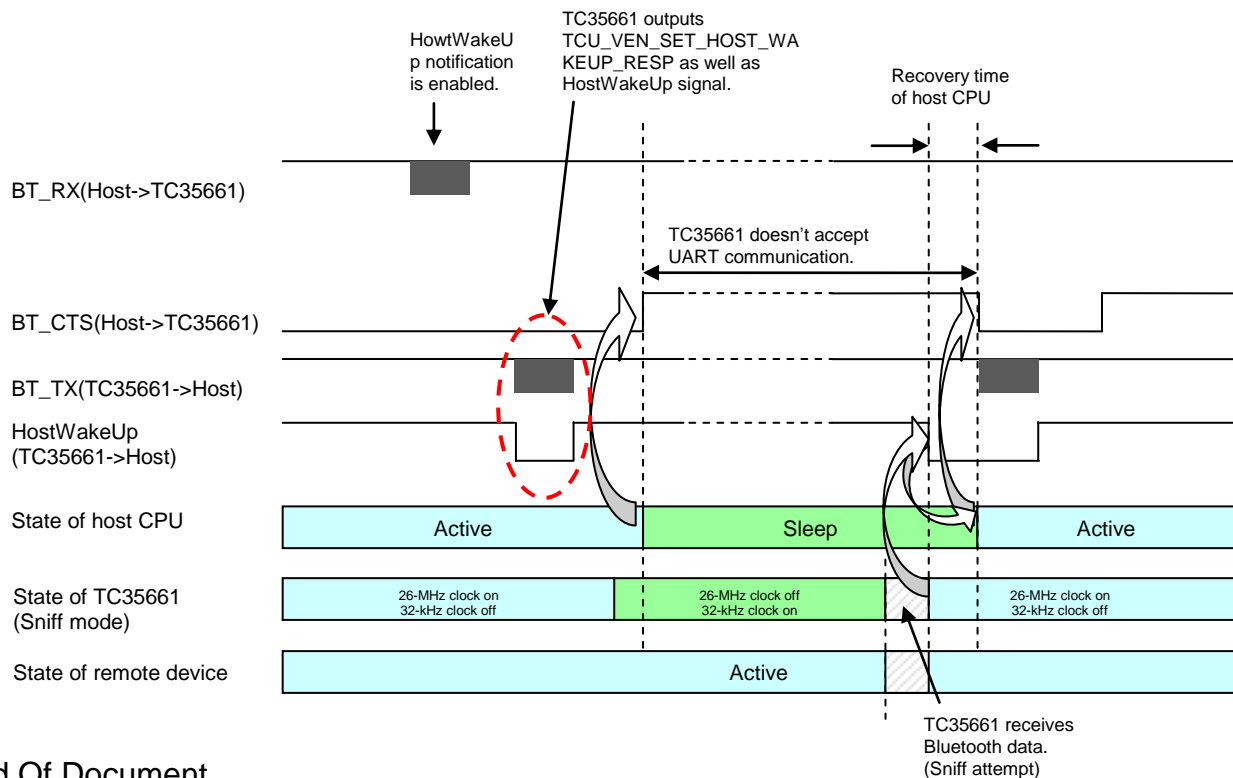
### 1.8.1. General Description

TCU\_VEN\_SET\_HOST\_WAKEUP\_NOTIFICATION\_REQ command enables HostWakeup signal on GPIO4, HostWakeup signal wakes up host CPU from sleep mode.

Both host CPU and TC35661 can enter sleep mode to reduce the system power consumption.

### 1.8.2. Sequence Example

- 1) Send TCU\_VEN\_SET\_HOST\_WAKEUP\_NOTIFICATION\_REQ(Enable) command from host CPU.
- 2) TC35661 sends TCU\_VEN\_SET\_HOST\_WAKEUP\_RESP to host CPU as well as HostWakeup signal simultaneously
- 3) Host CPU needs to turn BT\_CTS to high before entering sleep mode.
- 4) TC35661 is waked up by receiving data from remote device.
- 5) TC35661 outputs HostWakeup signal to host CPU.
- 6) Host CPU is waked up by receiving HostWakeup signal.
- 7) After host CPU wakes up, It turns BT\_CTS signal to low, and then TC35661 sends the received data.



End Of Document