

# User Manual

## DA14585/DA14531 HCI Commands

UM-B-146

### Abstract

*This document describes the HCI commands for the DA14585 and DA14531 devices which are related to the prod\_test project in the SDK.*

---

---

## Contents

<b>Abstract .....</b>	<b>1</b>
<b>Contents .....</b>	<b>2</b>
<b>Tables .....</b>	<b>3</b>
<b>1 Introduction.....</b>	<b>4</b>
<b>2 Overview of Supported Commands .....</b>	<b>5</b>
<b>3 Core Commands.....</b>	<b>7</b>
3.1 Reset.....	7
3.2 cont_pkt_tx.....	7
3.3 pkt_tx.....	8
3.4 start_pkt_rx .....	9
3.5 start_pkt_rx_stats.....	10
3.6 Stop Test.....	11
3.7 stop_pkt_rx_stats .....	11
3.8 Unmodulated OFF / TX / RX.....	12
3.9 start_cont_tx.....	13
3.10 stop_cont_tx.....	13
<b>4 Sleep Commands .....</b>	<b>15</b>
<b>5 XTAL Trim Commands.....</b>	<b>16</b>
<b>6 OTP Commands .....</b>	<b>18</b>
6.1 OTP Read .....	19
6.2 OTP Write (Not Supported in DA14531).....	21
<b>7 Various Commands.....</b>	<b>22</b>
7.1 Get Firmware Version .....	22
7.2 Change UART Pins.....	22
7.3 Custom Action.....	24
7.4 Sensor Test.....	25
7.5 GPIO Set.....	26
7.6 GPIO Read.....	27
7.7 UART Loop .....	28
7.8 UART Baud .....	28
7.9 XTAL32K Test.....	29
7.10 GPIO_WDOG.....	30
7.11 SLEEP Clock Select (Not Supported in DA14531).....	30
7.12 ADC VBAT Read.....	31
7.13 SET TX Power .....	32
7.14 Configure TEST (Only in DA14531).....	32
7.15 RD Tester Command (Not Supported in DA14531) .....	33
<b>8 Register Read/Write .....</b>	<b>35</b>
<b>Revision History .....</b>	<b>37</b>

**Tables**

Table 1: Supported Commands .....	5
Table 2: Reset .....	7
Table 3: cont_pkt_tx .....	7
Table 4: pkt_tx .....	8
Table 5: start_pkt_rx .....	9
Table 6: start_pkt_rx_stats .....	10
Table 7: stopstest .....	11
Table 8: stop_pkt_rx_stats .....	11
Table 9: Unmodulated OFF / TX / RX .....	12
Table 10: start_cont_tx .....	13
Table 11: stop_cont_tx .....	13
Table 12: Sleep None/Extended /Deep .....	15
Table 13: xtrim rd/wr/en/dis/inc/dec .....	16
Table 14: otp wr_xtrim/rd_xtrim/wr_bdaddr/rd_bdaddr .....	18
Table 15: otp_read .....	19
Table 16: otp_write .....	21
Table 17: hci_firmware_version_get .....	22
Table 18: hci_change_uart_pins_action .....	22
Table 19: hci_custom_action .....	24
Table 20: hci_sensor_test .....	25
Table 21: GPIO_set .....	26
Table 22: GPIO_read .....	27
Table 23: UART_loop .....	28
Table 24: UART_baud .....	28
Table 25: Ext32K_test .....	29
Table 26: GPIO_WDOG .....	30
Table 27: SLEEP_CLK_SEL (Not Supported in DA14531) .....	30
Table 28: ADC_VBAT (Only in DA14531) .....	31
Table 29: SET_TX_POWER (Only in DA14531) .....	32
Table 30: CONFIGURE_TEST (Only in DA14531) .....	32
Table 31: RD_TESTER_COMMAND (Not Supported in DA14531) .....	33
Table 32: read_reg32 / write_reg32 / read_reg16 / write_reg16 .....	35

## 1 Introduction

This application note describes all vendor specific HCI commands related to the `prod_test` project in the SDK. `<sdk_root_directory>\projects\target_apps\prod_test`. The `prod_test` application implements commands for production test purpose.

All the standard HCI commands are part of the Bluetooth specification (c.f [Core\\_v5.2](#), Volume 4 part E, section 7)

## 2 Overview of Supported Commands

**Table 1: Supported Commands**

SUPPORTED COMMANDS						
Definition	Opcode	IC Availability			Tab	Notes
		585	586	531		
HCI_CMD_OPCODE_CONT_PKT_TX	0x201E	yes	yes	yes	Core Commands	
HCI_CMD_OPCODE_START_PKT_RX	0x201D	yes	yes	yes	Core Commands	
HCI_CMD_OPCODE_STOPTEST	0x201F	yes	yes	yes	Core Commands	
HCI_CMD_OPCODE_RESET	0x0C03	yes	yes	yes	Core Commands	
HCI_CUSTOM_ACTION_CMD_OPCODE	0xFE00	yes	yes	yes	Various Commands	
HCI_SLEEP_TEST_CMD_OPCODE	0xFE01	yes	yes	yes	Sleep	
HCI_XTAL_TRIM_CMD_OPCODE	0xFE02	yes	yes	yes	XTAL Trim	
HCI_OTP_RW_CMD_OPCODE	0xFE03	yes	yes	yes	OTP	
HCI_OTP_READ_CMD_OPCODE	0xFE04	yes	yes	yes	OTP	
HCI_OTP_WRITE_CMD_OPCODE	0xFE05	yes	yes	no	OTP	
HCI_REGISTER_RW_CMD_OPCODE	0xFE06	yes	yes	yes	Register RW	
HCI_AUDIO_TEST_CMD_OPCODE	0xFE07	no	no	no	N/A	Reserved Opcode
HCI_FIRMWARE_VERSION_GET_CMD_OPCODE	0xFE08	yes	yes	yes	Various Commands	
HCI_CHANGE_UART_PINS_ACTION_CMD_OPCODE	0xFE09	yes	yes	yes	Various Commands	
HCI_RDTESTER_CMD_OPCODE	0xFE0A	yes	yes	no	Various Commands	
HCI_TX_TEST_CMD_OPCODE	0xFE0B	yes	yes	yes	Core Commands	
HCI_START_PROD_RX_TEST_CMD_OPCODE	0xFE0C	yes	yes	yes	Core Commands	
HCI_END_PROD_RX_TEST_CMD_OPCODE	0xFE0D	yes	yes	yes	Core Commands	
HCI_UNMODULATED_ON_CMD_OPCODE	0xFE0E	yes	yes	yes	Core Commands	

SUPPORTED COMMANDS						
HCI_TX_START_CONTINUE_TEST_CMD_OPCODE	0xFE0F	yes	yes	yes	Core Commands	
HCI_TX_END_CONTINUE_TEST_CMD_OPCODE	0xFE10	yes	yes	yes	Core Commands	
HCI_SENSOR_TEST_CMD_OPCODE	0xFE11	yes	yes	yes	Various Commands	
HCI_GPIO_SET_CMD_OPCODE	0xFE12	yes	yes	yes	Various Commands	
HCI_GPIO_READ_CMD_OPCODE	0xFE13	yes	yes	yes	Various Commands	
HCI_UART_LOOP_CMD_OPCODE	0xFE14	yes	yes	yes	Various Commands	
HCI_UART_BAUD_CMD_OPCODE	0xFE15	yes	yes	yes	Various Commands	
HCI_EXT32KHz_TEST_CMD_OPCODE	0xFE16	yes	yes	yes	Various Commands	
HCI_GPIO_WD_CMD_OPCODE	0xFE17	yes	yes	yes	Various Commands	
HCI_SLEEP_CLK_SEL_CMD_OPCODE	0xFE18	yes	yes	no	Various Commands	
HCI_RANGE_EXT_EN_CMD_OPCODE	0xFE19	no	no	no	N/A	Reserved Opcode
HCI_ADC_VBAT_CMD_OPCODE	0xFE1A	no	no	yes	Various Commands	
HCI_SET_TX_POWER_CMD_OPCODE	0xFE1B	no	no	yes	Various Commands	
HCI_CONFIGURE_TEST_MODE_CMD_OPCODE	0xFE1C	no	no	yes	Various Commands	
HCI_RESET_MODE_CMD_OPCODE	0xFE1E	yes	yes	yes	Various Commands	

### 3 Core Commands

#### 3.1 Reset

Table 2: Reset

reset			
<b>Command Description</b>	Perform reset		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x03	
	Command Opcode MSB	0x0C	
	Parameter Length	0x00	
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x04	
	Num_HCI_Command_Packets	0xXX	
	Command_Opcode LSB	0x03	
	Command_Opcode MSB	0x0c	
Status	0x00 - 0xFF	0x00: Reset command succeeded, was received and will be executed. 0x01-0xFF: Reset command failed. See Volume 2, Part D -Error Codes in Bluetooth 4.0 specification for a list of error codes and descriptions.	

#### 3.2 cont\_pkt\_tx

Table 3: cont\_pkt\_tx

cont_pkt_tx			
<b>Command Description</b>	Continuous packet transmission		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x1E	
	Command Opcode MSB	0x20	
	Parameter Length	0x03	
	Frequency	0x00 - 0x27	= (F – 2402) / 2, where F ranges from 2402 MHz to 2480 MHz.
	Data Length	0x01 - 0x25	Length in bytes of payload data in each packet

cont_pkt_tx			
	Payload Type	0x00 - 0x07	0x00: Pseudo-Random bit sequence 9 0x01: Pattern of alternating bits '11110000' 0x02: Pattern of alternating bits '10101010' 0x03: Pseudo-Random bit sequence 15 0x04: Pattern of All '1' bits 0x05: Pattern of All '0' bits 0x06: Pattern of alternating bits '00001111' 0x07: Pattern of alternating bits '0101'
Return Message	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x04	
	Num_HCI_Command_Packets	0xFF	
	Command_Opcode_LSB	0x1E	
	Command_Opcode_MSB	0x20	
	Status	0x00 - 0xFF	0x00: command succeeded. 0x01 – 0xFF: command failed. See Volume 2, Part D -Error Codes in Bluetooth 4.0 specification for a list of error codes and descriptions.

### 3.3 pkt\_tx

Table 4: pkt\_tx

pkt_tx			
<b>Command Description</b>	Packet transmission		
Command Format	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x0B	
	Command Opcode MSB	0xFE	
	Parameter Length	0x05	
	Frequency	0x00 - 0x27	= (F – 2402) / 2, where F ranges from 2402 MHz to 2480 MHz.
	Data Length	0x01-0x25	Length in bytes of payload data in each packet



pkt_tx			
	Payload Type	0x00 - 0x07	0x00: Pseudo-Random bit sequence 9 0x01: Pattern of alternating bits '11110000' 0x02: Pattern of alternating bits '10101010' 0x03: Pseudo-Random bit sequence 15 0x04: Pattern of All '1' bits 0x05: Pattern of All '0' bits 0x06: Pattern of alternating bits '00001111' 0x07: Pattern of alternating bits '0101'
	Number of packets to receive LSB	0xXX	
	Number of packets to receive MSB	0xXX	
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0F	
	Parameter Length	0x04	
	Status	0x00	
	Num_HCI_Command_Packets	0xXX	
	Command_Opcode LSB	0x0B	
	Command_Opcode MSB	0xFE	
<b>Message returned when transmission is completed</b>	Byte Description	Value	Notes
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x03	
	Num_HCI_Command_Packets	0xXX	
	Command_Opcode LSB	0x0B	
	Command_Opcode MSB	0xFE	

### 3.4 start\_pkt\_rx

Table 5: start\_pkt\_rx

start_pkt_rx			
<b>Command Description</b>	Start packet reception		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x1D	
	Command Opcode MSB	0x20	
	Parameter Length	0x01	
	Frequency	0x00 - 0x27	= (F - 2402) / 2, where F ranges from 2402 MHz to 2480 MHz.

start_pkt_rx			
Return Message	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x04	
	Num_HCI_Command_Packets	0xFF	
	Command_Opcode LSB	0x1D	
	Command_Opcode MSB	0x20	
	Status	0x00 - 0xFF	0x00: Command succeeded. 0x01-0xFF: Command failed. See Volume 2, Part D -Error Codes in Bluetooth 4.0 specification for a list of error codes and descriptions.

### 3.5 start\_pkt\_rx\_stats

Table 6: start\_pkt\_rx\_stats

start_pkt_rx_stats			
<b>Command Description</b>	Start packet reception, it also gathers statistics		
<b>Notes</b>	works with 0xFE0D		
Command Format	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x0C	
	Command Opcode MSB	0xFE	
	Parameter Length	0x01	
	Frequency	0x00 - 0x27	= (F - 2402) / 2, where F ranges from 2402 MHz to 2480 MHz.
Return Message	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x03	
	Num_HCI_Command_Packets	0xFF	
	Command_Opcode LSB	0x0C	
	Command_Opcode MSB	0xFE	

### 3.6 Stop Test

Table 7: stoptest

stoptest			
<b>Command Description</b>	Stop test		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x1F	
	Command Opcode MSB	0x20	
	Parameter Length	0x00	
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x06	
	Num_HCI_Command_Packets	0xFF	
	Command_Opcode LSB	0x1F	
	Command_Opcode MSB	0x20	
	Status	0x00 - 0xFF	0x00: Command succeeded. 0x01-0xFF: Command failed. See Volume 2, Part D -Error Codes in Bluetooth 4.0 specification for a list of error codes and descriptions.
	Number of packets received LSB	0xFF	
	Number of packets received MSB	0xFF	

### 3.7 stop\_pkt\_rx\_stats

Table 8: stop\_pkt\_rx\_stats

stop_pkt_rx_stats			
<b>Command Description</b>	Stop packet reception, it also returns statistics gathered during the test		
<b>Notes</b>	Works with 0xFE0C		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x0D	
	Command Opcode MSB	0xFE	
	Parameter Length	0x00	
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x0B	
	Num_HCI_Command_Packets	0xFF	
	Command_Opcode LSB	0X0D	

stop_pkt_rx_stats			
	Command_Opcode MSB	0xFE	
	Number of received packets LSB	0xFF	
	Number of received packets MSB	0xFF	
	Number of received packets with sync errors LSB	0xFF	
	Number of received packets with sync errors MSB	0xFF	
	Number of received packets with CRC errors LSB	0xFF	
	Number of received packets with CRC errors MSB	0xFF	
	RSSI LSB	0xFF	RSSI value is converted to dBm according to the following formula: DA14585: dBm = (479 * RSSI) / 1000 - 112.5 DA14531: dBm = 0.498 * RSSI - 127 (The range of valid DA14531 RSSI values is 40 to 230. Any values lower than 40 should be increased to 40. Any values higher than 230 should be lowered to 230.)
	RSSI MSB	0xFF	

### 3.8 Unmodulated OFF / TX / RX

Table 9: Unmodulated OFF / TX / RX

Unmodulated OFF / TX / RX			
<b>Command Description</b>	Unmodulated transmission/reception		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x0E	
	Command Opcode MSB	0xFE	
	Parameter Length	0x02	
	Operation	0x4F, 0x52 or 0x54	0x4F: OFF 0x54: unmodulated TX 0x52: unmodulated RX
	Frequency	0x00 - 0x27	= (F - 2402) / 2, where F ranges from 2402 MHz to 2480 MHz.
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x03	

Unmodulated OFF / TX / RX			
	Num_HCI_Command_Packets	0xXX	
	Command_Opcode LSB	0x0E	
	Command_Opcode MSB	0xFE	

### 3.9 start\_cont\_tx

Table 10: start\_cont\_tx

start_cont_tx			
<b>Command Description</b>	Start continuous transmission		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x0F	
	Command Opcode MSB	0xFE	
	Parameter Length	0x02	
	Frequency	0x00 - 0x27	= $(F - 2402) / 2$ , where F ranges from 2402 MHz to 2480 MHz.
	Payload Type	0x00 - 0x07	0x00: Pseudo-Random bit sequence 9 0x01: Pattern of alternating bits '11110000' 0x02: Pattern of alternating bits '10101010' 0x03: Pseudo-Random bit sequence 15 0x04: Pattern of All '1' bits 0x05: Pattern of All '0' bits 0x06: Pattern of alternating bits '00001111' 0x07: Pattern of alternating bits '0101'
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x03	
	Num_HCI_Command_Packets	0xXX	
	Command_Opcode LSB	0x0F	
	Command_Opcode MSB	0xFE	

### 3.10 stop\_cont\_tx

Table 11: stop\_cont\_tx

stop_cont_tx			
<b>Command Description</b>	Stop continuous transmission		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x10	

stop_cont_tx			
	Command Opcode MSB	0xFE	
	Parameter Length	0x00	
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Length	0x03	
	Num_HCI_Command_Packets	0xFF	
	Command_Opcode LSB	0x10	
	Command_Opcode MSB	0xFE	

## 4 Sleep Commands

Table 12: Sleep None/Extended /Deep

sleep none/extended /deep			
<b>Command Description</b>	Set the sleep state of the device		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x01	
	Command Opcode MSB	0xFE	
	Parameter Length	0x03	
	Sleep Mode	0x00 - 0x02	0x00: active mode 0x01: extended sleep 0x02: deep sleep
	Minutes to sleep	0x00- 0xFF	If both minutes = 0 and seconds = 0 then the device sleeps forever.
	Seconds to sleep	0x00- 0xFF	
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0F	
	Parameter Length	0x04	
	Status	0x00	
	Num_HCI_Command_Packets	0xFF	
	Command_Opcode LSB	0x01	
Command_Opcode MSB	0xFE		

## 5 XTAL Trim Commands

Table 13: xtrim rd/wr/en/dis/inc/dec

xtrim rd/wr/en/dis/inc/dec			
<b>Command Description</b>	Perform XTAL calibration operation		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x02	
	Command Opcode MSB	0xFE	
	Parameter Length	0x03	
	Operation	0x00 - 0x07	0x00: read trim val 0x01: write trim val 0x02: enable output xtal on P05 0x03: increase trim value by delta 0x04: decrease trim value by delta 0x05: disable XTAL output on P05 0x06: auto calibration test 0x07: auto calibration test (+ Burn calculated value to OTP, Burn applies only for DA14585/6)
	Trim value or delta LSB	0x00-0xFF	trim value L.SB when operation=1 delta value LSB when operation=3,4 GPIO* when operation = 6,7 0x00 otherwise.  *GPIO Px_y is encoded as x*10 + y. E.g. P1_5 is encoded as 15 (0x0F). If the GPIO value equals to 0xFE then the firmware automatically finds the UART RX pin and set it as an input pulse pin.
Trim value or delta MSB	0x00-0xFF	trim value MSB when operation=1 delta value MSB when operation=3,4 XTAL type* when operation = 6,7 0x00 otherwise. *XTAL type is 0: XTAL16M or 1: XTAL32M.	
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x05	
	Num_HCI_Command_Packets	0xXX	
	Command_Opcode LSB	0x02	
	Command_Opcode MSB	0xFE	
	Trim value LSB	0xXX	CLK_FREQ_TRIM_REG value for operation=0



xtrim rd/wr/en/dis/inc/dec			
	Trim value MSB	0xXX	status code* for operation=6,7 0x0000 otherwise. *XTAL trim value calibration returns zero on success. A non-zero value indicates failure. 1: XTAL_TRIM_OUT_OF_RANGE 2: XTAL_TRIM_FREQ_CAL_NOT_CONNECTED 3: XTAL_TRIM_OTP_WRITE_FAILED 4: PARAMS_ERROR (wrong GPIO) 5: PARAMS_ERROR (wrong XTAL type)

## 6 OTP Commands

Table 14: otp wr\_xtrim/rd\_xtrim/wr\_bdaddr/rd\_bdaddr

otp wr_xtrim/rd_xtrim/wr_bdaddr/rd_bdaddr						
<b>Command Description</b>	Read or write BD address and XTAL Trim fields in OTP header					
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>			
	HCI Command Packet	0x01				
	Command Opcode LSB	0x03				
	Command Opcode MSB	0xFE				
	Parameter Length	0x07				
	Operation	0xXX	0x00	0x01	0x02	0x03
			read XTAL16M trim value OTP header field (not supported in DA14531)	write XTAL16M trim value OTP header field (not supported in DA14531)	read BD address OTP header field	write BD address OTP header field
	data[0]	0xXX	(Not used)	trim value LSB	(Not used)	bdaddr[0] (LSB)
	data[1]	0xXX	(Not used)	trim value MSB	(Not used)	bdaddr[1]
	data[2]	0xXX	(Not used)	(Not used)	(Not used)	bdaddr[2]
	data[3]	0xXX	(Not used)	(Not used)	(Not used)	bdaddr[3]
	data[4]	0xXX	(Not used)	(Not used)	(Not used)	bdaddr[4]
	data[5]	0xXX	(Not used)	(Not used)	(Not used)	bdaddr[5]
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>			
	HCI Event Packet	0x04				
	Event Code	0x0E				
	Parameter Length	0x0A				
	Num_HCI_Command_Packets	0xXX				
	Command_Opcode LSB	0x03				
	Command_Opcode MSB	0xFE				
	Operation	0xXX	0x00	0x01	0x02	0x03
			read XTAL16M trim value (not supported in DA14531)	write XTAL16M trim value (not supported in DA14531)	read BD address	write BD address

otp_wr_xtrim/rd_xtrim/wr_bdaddr/rd_bdaddr						
	data[0]	0xXX	trim value LSB	(Not used)	bdaddr[0] (LSB)	(Not used)
	data[1]	0xXX	trim value MSB	(Not used)	bdaddr[1]	(Not used)
	data[2]	0xXX	(Not used)	(Not used)	bdaddr[2]	(Not used)
	data[3]	0xXX	(Not used)	(Not used)	bdaddr[3]	(Not used)
	data[4]	0xXX	(Not used)	(Not used)	bdaddr[4]	(Not used)
	data[5]	0xXX	(Not used)	(Not used)	bdaddr[5]	(Not used)

## 6.1 OTP Read

Table 15: otp\_read

otp_read			
<b>Command Description</b>	Read a field in OTP		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x04	
	Command Opcode MSB	0xFE	
	Parameter Length	0x03	
	Start word address LSB	0xXX	The address must be word aligned
	Start word address MSB	0xXX	
	# words to read	0xXX	min = 1 max = 60
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0xXX	0x05 + 4 * (# words returned)
	Num_HCI_Command_Packets	0xXX	
	Command_Opcode LSB	0x04	
	Command_Opcode MSB	0xFE	
	Status	0x00 or 0x12	0x00: command succeeded 0x12: Invalid HCI parameter
	# words returned (=n)	0xXX	(# words returned)
	word 1 byte 0 (LSB)	0xXX	
	word 1 byte 1	0xXX	
	word 1 byte 2	0xXX	
	word 1 byte 3	0xXX	

otp_read			
	...	...	
	word n byte 0 (LSB)	0xXX	
	word n byte 1	0xXX	
	word n byte 2	0xXX	
	word n byte 3	0xXX	

## 6.2 OTP Write (Not Supported in DA14531)

Table 16: otp\_write

otp_write (not supported in DA14531)			
<b>Command Description</b>	Write a field in OTP		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x05	
	Command Opcode MSB	0xFE	
	Parameter Length	0xXX	0x03 + 4 * (# words to write)
	Start word address LSB	0xXX	the address must be word aligned, lower than 0x8000 and the following must hold
	Start word address MSB	0xXX	start_address + 4 * number_of_words_to_write < 0x8000
	# words to write (= n)	0xXX	min = 1 max = 60
	word 1 byte 0 (LSB)	0xXX	
	word 1 byte 1	0xXX	
	word 1 byte 2	0xXX	
	word 1 byte 3	0xXX	
	...	0xXX	
	word n byte 0 (LSB)	0xXX	
	word n byte 1	0xXX	
	word n byte 2	0xXX	
word n byte 3	0xXX		
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x05	
	Num_HCI_Command_Packets	0xXX	
	Command_Opcode LSB	0x05	
	Command_Opcode MSB	0xFE	
	Status	0x00 or 0x12	0x00: command succeeded 0x12: Invalid HCI parameter
	# words written	0xXX	

## 7 Various Commands

### 7.1 Get Firmware Version

Table 17: hci\_firmware\_version\_get

hci_firmware_version_get			
<b>Command Description</b>	Retrieve the BLE and application version of the firmware		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x08	
	Command Opcode MSB	0xFE	
	Parameter Length	0x00	
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x45	
	Num_HCI_Command_Packets	0xFF	
	Command_Opcode LSB	0x08	
	Command_Opcode MSB	0xFE	
	BLE_version_length	0xFF	min = 0 max = 32
	Application_version_length (Max value 32).	0xFF	min = 0 max = 32
	BLE_common_firmware_version (32 bytes).	""	32-byte string containing the BLE common firmware version.
BLE_application_firmware_version (32 bytes).	""	32-byte string containing the BLE application firmware version.	

### 7.2 Change UART Pins

Table 18: hci\_change\_uart\_pins\_action

hci_change_uart_pins_action			
<b>Command Description</b>	Change the UART pins		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x09	
	Command Opcode MSB	0xFE	
	Parameter Length	0x04	
	tx_port	0xFF	GPIO Px_y is encoded as x*10 + y. E.g. P1_5 is encoded as 15 (0x0F).
	tx_pin	0xFF	GPIO Px_y is encoded as x*10 + y. E.g. P1_5 is encoded as 15 (0x0F).

hci_change_uart_pins_action			
	rx_port	0xXX	GPIO Px_y is encoded as $x*10 + y$ . E.g. P1_5 is encoded as 15 (0x0F).
	rx_pin	0xXX	GPIO Px_y is encoded as $x*10 + y$ . E.g. P1_5 is encoded as 15 (0x0F).
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x05	
	Num_HCI_Command_Packets	0xXX	
	Command_Opcode LSB	0x09	
	Command_Opcode MSB	0xFE	
	Status LSB	0xXX	0x00: Command succeeded. 0x01: Commands failed
	Status MSB	0x00	

### 7.3 Custom Action

Table 19: hci\_custom\_action

hci_custom_action			
<b>Command Description</b>	Run a custom action. These actions are user-defined inside the code.		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x00	
	Command Opcode MSB	0xFE	
	Parameter Length	0x01	
	Custom action	0XX	User defined inside the firmware
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x07	
	Num_HCI_Command_Packets	0XX	
	Command_Opcode LSB	0x00	
	Command_Opcode MSB	0xFE	
	Return Data LSB	0XX	By default, echoes back the Custom action byte received on success Can be used by the user defined operation
	Return Data	0XX	Can be used by the user defined operation
	Return Data	0XX	Can be used by the user defined operation
	Return Data	0XX	Can be used by the user defined operation



## 7.4 Sensor Test

Table 20: hci\_sensor\_test

hci_sensor_test			
<b>Command Description</b>	Perform basic communication test on an attached sensor		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x11	
	Command Opcode MSB	0xFE	
	Parameter Length	0x11	
	Interface	0x00 - 0x01	0x00: Sensor connected to SPI bus 0x01: Sensor connected to I2C bus
	Read/Write	0x00 - 0x01	0x00: Read 0x01: Write
	spi_clk_port or i2c_scl_port	0xXX	DA14585/6: Ports 0x00 to 0x03 are available DA14531: This should always be 0x00
	spi_clk_pin or i2c_scl_pin	0xXX	DA14585/6: Port0: 0x00 to 0x07 Port1: 0x00 to 0x05 Port2: 0x00 to 0x09 Port3: 0x00 to 0x07 DA14531: 0x00 to 0x0B
	spi_di_port or i2c_sda_port	0xXX	DA14585/6: Ports 0x00 to 0x03 are available DA14531: This should always be 0x00
	spi_di_pin or i2c_sda_pin	0xXX	DA14585/6: Port0: 0x00 to 0x07 Port1: 0x00 to 0x05 Port2: 0x00 to 0x09 Port3: 0x00 to 0x07 DA14531: 0x00 to 0x0B
	spi_do_port	0xXX	DA14585/6: Ports 0x00 to 0x03 are available DA14531: This should always be 0x00
	spi_do_pin	0xXX	DA14585/6: Port0: 0x00 to 0x07 Port1: 0x00 to 0x05 Port2: 0x00 to 0x09 Port3: 0x00 to 0x07 DA14531: 0x00 to 0x0B
	spi_cs_port	0xXX	DA14585/6: Ports 0x00 to 0x03 are available DA14531: This should always be 0x00
	spi_cs_pin	0xXX	DA14585/6: Port0: 0x00 to 0x07 Port1: 0x00 to 0x05 Port2: 0x00 to 0x09 Port3: 0x00 to 0x07 DA14531: 0x00 to 0x0B
Register address	0xXX	Sensor register address	
Register data to write	0xXX	Data to write to the sensor register if Read/Write = 0x01	

hci_sensor_test			
	I2C slave address	0xXX	The sensor I2C slave address used if Interface = 0x01
	int_gpio_check	0xXX	0 = Do nothing. 1 = Set the following GPIO to input pull-down after the interface (SPI or I2C) has been initialized.
	int_port	0xXX	DA14585/6: Ports 0x00 to 0x03 are available DA14531: This should always be 0x00
	int_pin	0xXX	DA14585/6: Port0: 0x00 to 0x07 Port1: 0x00 to 0x05 Port2: 0x00 to 0x09 Port3: 0x00 to 0x07 DA14531: 0x00 to 0x0B
	Pins voltage level	0xXX	0 = 3.3 V 1 = 1.8 V
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x05	
	Num_HCI_Command_Packets	0xXX	
	Command_Opcode LSB	0x11	
	Command_Opcode MSB	0xFE	
	Sensor register data or INT GPIO level	0xXX	Byte read from address specified in byte "Register address" shown in the command format above, or the INT GPIO level (high = 0x01 or low = 0x00) if int_gpio_check = 0x01.
Error	0x00 or 0xFF	0x00 = Command succeeded - Sensor register data or INT GPIO level data are valid. 0xFF = Command error - Sensor register data or INT GPIO level data are invalid.	

### 7.5 GPIO Set

Table 21: GPIO\_set

GPIO_set			
<b>Command Description</b>	Set the state of a GPIO		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x12	
	Command Opcode MSB	0xFE	
	Parameter Length	0x06	

GPIO_set			
	GPIO pad	0xXX	GPIO Px_y is encoded as x*10 + y. E.g. P1_5 is encoded as 15 (0x0F).
	mode	0x00 - 0x03	0: mode = INPUT 1: mode = INPUT_PULLUP 2: mode = INPUT_PULLDOWN 3: mode = OUTPUT
	Voltage level	0x00 - 0x01	0 = 3.3 V 1 = 1.8 V
	state	0x00 - 0x01	0 = low 1 = high
	pwm	0xXX	0 = No PWM otherwise = frequency in kHz
	duty_cycle	0x00 - 0x64	duty_cycle percentage
Return Message	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x04	
	Num_HCI_Command_Packets	0xXX	
	Command_Opcode_LSB	0x12	
	Command_Opcode_MSB	0xFE	
	Error	0x00 or 0xFF	0x00 = Command succeeded 0xFF = Command error

## 7.6 GPIO Read

Table 22: GPIO\_read

GPIO_read			
<b>Command Description</b>	Read the state of a GPIO		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x13	
	Command Opcode MSB	0xFE	
	Parameter Length	0x01	
	GPIO pad	0xXX	GPIO Px_y is encoded as x*10 + y. E.g. P1_5 is encoded as 15 (0x0F).
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	

GPIO_read			
	Parameter Length	0x04	
	Num_HCI_Command_Packets	0xXX	
	Command_Opcode_LSB	0x13	
	Command_Opcode_MSB	0xFE	
	Data	0x00, 0x01 or 0xFF	0x00 = high GPIO state 0x01 = low GPIO state 0xFF = Command error

### 7.7 UART Loop

Table 23: UART\_loop

UART_loop			
<b>Command Description</b>	Test UART noise existence using an echo test		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x14	
	Command Opcode MSB	0xFE	
	Parameter Length	0x64	
	Data[100]	0xXX	Input data buffer
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x67	
	Num_HCI_Command_Packets	0xXX	
	Command_Opcode_LSB	0x14	
	Command_Opcode_MSB	0xFE	
	Data[100]	0x00 - 0x05	Echo of input data buffer

### 7.8 UART Baud

Table 24: UART\_baud

UART_baud			
<b>Command Description</b>	Set the UART baud rate		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x15	
	Command Opcode MSB	0xFE	
	Parameter Length	0x01	

UART_baud			
	Data	0xXX	0x00: UART_BAUDRATE_9K6 0x01: UART_BAUDRATE_19K2 0x02: UART_BAUDRATE_57K6 0x03: UART_BAUDRATE_115K2 0x04: UART_BAUDRATE_1M 0x05: UART_BAUDRATE_38K4
Return Message	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x04	
	Num_HCI_Command_Packets	0xXX	
	Command_Opcode LSB	0x15	
	Command_Opcode MSB	0xFE	
	Error	0xXX	

### 7.9 XTAL32K Test

Table 25: Ext32K\_test

Ext32K_test			
<b>Command Description</b>	Check the accuracy of the XTAL32K		
Command Format	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x16	
	Command Opcode MSB	0xFE	
	Parameter Length	0x00	
Return Message	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x04	
	Num_HCI_Command_Packets	0xXX	
	Command_Opcode LSB	0x16	
	Command_Opcode MSB	0xFE	
	Error	0x00 or 0xFF	0x00 = Command succeeded 0xFF = Command error

## 7.10 GPIO\_WDOG

Table 26: GPIO\_WDOG

GPIO_WDOG			
<b>Command Description</b>	Continuously toggle a GPIO to be used as a WDOG indication on an external mcu		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x17	
	Command Opcode MSB	0xFE	
	Parameter Length	0x02	
	GPIO pad	0xXX	GPIO Px_y is encoded as x*10 + y. E.g. P1_5 is encoded as 15 (0x0F).
	Voltage level	0xXX	0 = 3.3 V 1 = 1.8 V
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x04	
	Num_HCI_Command_Packets	0xXX	
	Command_Opcode_LSB	0x17	
	Command_Opcode_MSB	0xFE	
	Error	0x00 or 0xFF	0x00 = Command succeeded 0xFF = Command error

## 7.11 SLEEP Clock Select (Not Supported in DA14531)

Table 27: SLEEP\_CLK\_SEL (Not Supported in DA14531)

SLEEP_CLK_SEL (Not Supported in DA14531)			
<b>Command Description</b>	Select the clock that will be used during sleep		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x18	
	Command Opcode MSB	0xFE	
	Parameter Length	0x01	
	sleep_clk	0x00 or 0x01	0x00 = RCX20 0x01 = XTAL32K
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	

SLEEP_CLK_SEL (Not Supported in DA14531)			
	Event Code	0x0E	
	Parameter Length	0x04	
	Num_HCI_Command_Packets	0xXX	
	Command_Opcode LSB	0x18	
	Command_Opcode MSB	0xFE	
	Error	0x00 or 0xFF	0x00 = Command succeeded 0xFF = Command error

## 7.12 ADC VBAT Read

Table 28: ADC\_VBAT (Only in DA14531)

ADC_VBAT (Only in DA14531)			
<b>Command Description</b>	Get the VBAT voltage using the ADC		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x1A	
	Command Opcode MSB	0xFE	
	Parameter Length	0x01	
	DC-DC mode: BOOST	0xXX	
	DC-DC mode: BUCK	0xXX	
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x05	
	Num_HCI_Command_Packets	0xXX	
	Command_Opcode LSB	0x1A	
	Command_Opcode MSB	0xFE	
	vbat level LSB	0xXX	= vbat_level * (3600/2047)
	vbat level MSB	0xXX	

7.13 SET TX Power

Table 29: SET\_TX\_POWER (Only in DA14531)

SET_TX_POWER (Only in DA14531)			
<b>Command Description</b>	Set radio tx power		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x1B	
	Command Opcode MSB	0xFE	
	Parameter Length	0x01	
	rf_tx_pwr_lvl	0x01 to 0x0C	0x01: -19.5 dBm 0x02: -13.5 dBm 0x03: -10 dBm 0x04: -7 dBm 0x05: -5 dBm 0x06: -3.5 dBm 0x07: -2 dBm 0x08: -1 dBm 0x09: 0 dBm 0x0A: +1 dBm 0x0B: +1.5 dBm 0x0C: +2 dBm
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x04	
	Num_HCI_Command_Packets	0xXX	
	Command_Opcode LSB	0x1B	
	Command_Opcode MSB	0xFE	
Error	0x00	0x00 = Command succeeded	

7.14 Configure TEST (Only in DA14531)

Table 30: CONFIGURE\_TEST (Only in DA14531)

CONFIGURE_TEST (Only in DA14531)			
<b>Command Description</b>	Set radio continuous mode		
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x1C	
	Command Opcode MSB	0xFE	
Parameter Length	0x01		



CONFIGURE_TEST (Only in DA14531)			
	Data	0xXX	0x00=Radio LDOs in slotted mode 0x01=Radio LDOs in continuous mode. In buck mode VBAT_LOW is configured at 2.5V. In boost mode, VBAT_LOW supply must be >2.2 V
Return Message	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x04	
	Num_HCI_Command_Packets	0xXX	
	Command_Opcode LSB	0x1C	
	Command_Opcode MSB	0xFE	
	Error	0x00	0x00 = Command succeeded

### 7.15 RD Tester Command (Not Supported in DA14531)

Table 31: RD\_TESTER\_COMMAND (Not Supported in DA14531)

RD_TESTER_COMMAND (not supported in DA14531)			
<b>Command Description</b>	Control the CPLD located on the PLT HW		
Command Format	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Command Packet	0x01	
	Command Opcode LSB	0x0A	
	Command Opcode MSB	0xFE	
	Parameter Length	0x03	
	Operation	0xXX	0x00: RDTESTER_INIT 0x01: RDTESTER_UART_CONNECT 0x02: RDTESTER_UART_LOOPBACK 0x03: RDTESTER_VBAT_CNTRL 0x04: RDTESTER_VPP_CNTRL 0x05: RDTESTER_RST_PULSE 0x06: RDTESTER_UART_PULSE 0x07: RDTESTER_XTAL_PULSE 0x08: RDTESTER_PULSE_WIDTH
	Data LSB	0xXX	

RD_TESTER_COMMAND (not supported in DA14531)			
	Data MSB	0xXX	RDTESTER_INIT: N/A RDTESTER_UART_CONNECT: DUT_map RDTESTER_UART_LOOPBACK: DUT_port RDTESTER_VBAT_CNTRL: DUT_map RDTESTER_VPP_CNTRL: VPP_state RDTESTER_RST_PULSE: delay_ms RDTESTER_UART_PULSE: DUT_map RDTESTER_XTAL_PULSE: N/A RDTESTER_PULSE_WIDTH: length
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>
	HCI Event Packet	0x04	
	Event Code	0x0E	
	Parameter Length	0x03	
	Num_HCI_Command_Packets	0xXX	
	Command_Opcode LSB	0x0A	
	Command_Opcode MSB	0xFE	

## 8 Register Read/Write

Table 32: read\_reg32 / write\_reg32 / read\_reg16 / write\_reg16

read_reg32 / write_reg32 / read_reg16 / write_reg16						
<b>Command Description</b>	Read or Write a register					
<b>Command Format</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>			
	HCI Command Packet	0x01				
	Command Opcode LSB	0x06				
	Command Opcode MSB	0xFE				
	Parameter Length	0x09				
	Operation	0xXX	0x00	0x01	0x02	0x03
			read_reg32	write_reg32	read_reg16	write_reg16
	addr[0]	0xXX	register address byte 0 (LSB)			
	addr[1]	0xXX	register address byte 1			
	addr[2]	0xXX	register address byte 2			
	addr[3]	0xXX	register address byte 3			
	data[0]	0xXX	(Not used)	value byte 0 (LSB)	(Not used)	value byte 0 (LSB)
	data[1]	0xXX	(Not used)	value byte 1	(Not used)	value byte 1
	data[2]	0xXX	(Not used)	value byte 2	(Not used)	(Not used)
	data[3]	0xXX	(Not used)	value byte 3	(Not used)	(Not used)
<b>Return Message</b>	<b>Byte Description</b>	<b>Value</b>	<b>Notes</b>			
	HCI Event Packet	0x04				
	Event Code	0x0E				
	Parameter Length	0x09				
	Num_HCI_Command_Packets	0xXX				
	Command_Opcode LSB	0x06				
	Command_Opcode MSB	0xFE				
	Operation	0xXX	0x00	0x01	0x02	0x03
			read_reg32	write_reg32	read_reg16	write_reg16
	Reserved	0x00	0x00	0x00	0x00	0x00
data[0]	0xXX	value byte 0 (LSB)	(Not used)	value byte 0 (LSB)	(Not used)	
data[1]	0xXX	value byte 1	(Not used)	value byte 1	(Not used)	
data[2]	0xXX	value byte 2	(Not used)	(Not used)	(Not used)	

read_reg32 / write_reg32 / read_reg16 / write_reg16						
	data[3]	0xXX	value byte 3	(Not used)	(Not used)	(Not used)

## Revision History

Revision	Date	Description
1.2	14-Sep-2022	Sections 2, 7.12 and 7.13 updated.
1.1	25-Jan-2022	Updated logo, disclaimer, copyright.
1.0	10-Nov-2020	Initial Release

**Status Definitions**

Status	Definition
DRAFT	The content of this document is under review and subject to formal approval, which may result in modifications or additions.
APPROVED or unmarked	The content of this document has been approved for publication.

**RoHS Compliance**

Dialog Semiconductor's suppliers certify that its products are in compliance with the requirements of Directive 2011/65/EU of the European Parliament on the restriction of the use of certain hazardous substances in electrical and electronic equipment. RoHS certificates from our suppliers are available on request.

## IMPORTANT NOTICE AND DISCLAIMER

RENESAS ELECTRONICS CORPORATION AND ITS SUBSIDIARIES (“RENESAS”) PROVIDES TECHNICAL SPECIFICATIONS AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for developers skilled in the art designing with Renesas products. You are solely responsible for (1) selecting the appropriate products for your application, (2) designing, validating, and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. Renesas grants you permission to use these resources only for development of an application that uses Renesas products. Other reproduction or use of these resources is strictly prohibited. No license is granted to any other Renesas intellectual property or to any third party intellectual property. Renesas disclaims responsibility for, and you will fully indemnify Renesas and its representatives against, any claims, damages, costs, losses, or liabilities arising out of your use of these resources. Renesas' products are provided only subject to Renesas' Terms and Conditions of Sale or other applicable terms agreed to in writing. No use of any Renesas resources expands or otherwise alters any applicable warranties or warranty disclaimers for these products.

(Rev.1.0 Mar 2020)

### Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,  
Koto-ku, Tokyo 135-0061, Japan  
[www.renesas.com](http://www.renesas.com)

### Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:  
[www.renesas.com/contact/](http://www.renesas.com/contact/)

### Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.